AIAA FLIGHT SIMULATION TECHNOLOGIES CONFERENCE

A COLLECTION OF TECHNICAL PAPERS
LONG BEACH, CALIFORNIA
JUNE 16-18, 1981

AIAA at 50

Anniversary Celebration

A COLLECTION
OF
TECHNICAL PAPERS

AIAA FLIGHT SIMULATION TECHNOLOGIES CONFERENCE

June 16-18, 1981/Long Beach/California

Conference Committee

General Chairman

KEN DYDA
Rockwell International

Technical Program Co-Chairmen

VICTOR FACONTI
Singer Co.

JAMES L. COPELAND
NASA Langley Research Center

TABLE OF CONTENTS

LATE PAPERS

A-10A OPERATIONAL FLIGHT TRAINER SIMULATOR FLIGHT CONTROL SYSTEM AND AERODYNAMICS

81-0964

K.L. Johnson*
Aeronautical Systems Division
Wright-Patterson AFB, Dayton, Ohio
A.I. Portalatin** and P.W. Rinali+
Reflectone, Incorporated
Tampa, Florida

Abstract

The paper discusses the flight control aerodynamic and propulsive problems that occurred during development and test of the A-10A Trainer simulator and their solutions. The hardware and software design, and systems interface of these subsystems are also presented. General physical characteristics and capabilities of the A-10A aircraft and Trainer simulator are discussed. The paper proposes that a strong multidicipline technical linkage exists between computer systems, flight control loading hardware and electronics, and aerodynamics in order to build a real time, high fidelity, Operational Flight Trainer simulator.

Nomenclature

AI, AO	analog input, output
APU	auxiliary power unit
C_{m_0}	zero angle of attack pitching moment
C_{m_α}	pitching moment change with angle of attack
Count	.0001 of a unit
CRT	cathode ray tube
DCL	digital control loader
DI, DO	digital input, output
DMA	direct memory access
F_g	gross thrust
F_{NET}	net thrust
F_{RAM}	ram drag
H/W	hardware
Hz	cycles per second
ITT	interstage turbine temperature
m.a.c.	mean aerodynamic chord
N_g, $\dot{N}_g$	gas generator (core) RPM and rate of change of RPM
PLA	power lever angle/throttle position
P_{s3}	gas generator discharge pressure
RPM	revolutions per minute
SAS	stability augmentation system
w_f, $\dot{w}_f$	fuel flow and rate of change of fuel flow
xducers	transducers

A-10A Aircraft

The A-10A aircraft is a single place close-air support aircraft built by Fairchild Republic Company, Farmingdale, New York. It is powered by General Electric TF34-GE-100 engine having a maximum installed thrust of 9,000 pounds per engine. The maximum gross weight of the aircraft is 47,500 pounds. The engine is a high bypass turbofan with a single fan rotor, fourteen stage compressor and six turbine stages (See Fig. 1). It has a length of 53.3 feet, a wing span of 57.5 feet, and a wing area of 506 square feet. The wing airfoil is an NACA 6716 inboard of the landing gear pods and an NACA 6713 outboard. The wing contains four panel, three position flaps, aileron and aileron tab surfaces, and eight pylon weapon stations; three additional pylon stations are located on the fuselage. The ailerons consist of upper and lower panels which also function as speedbrakes when moved symmetrically. The empennage consists of twin verticals and rudders, a horizontal stabilizer, and elevators with elevator tab surfaces. The horizontal stabilizer and vertical stabilizer are NACA 64A013 airfoils. The armament system includes a 4,000 round per minute 30mm seven barrel gun. The flight control system is designed to operate with a single or dual hydraulic shut down; the latter case is called Manual Reversion. Without hydraulic power, roll control mechanically transfers from aileron surface control to aileron tab control by means of a roll tab shifter device near the control surfaces. Mechanical disconnect devices in both the pitch and roll control axes free the control stick to operate in one of two separate paths in both pitch and roll in the event of a mechanical linkage jam.

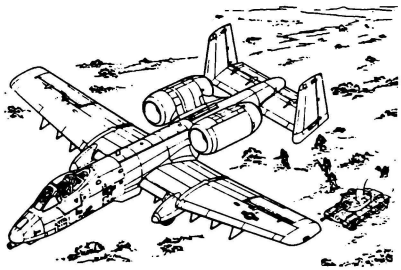


Fig. 1 A-10A Aircraft

A-10A Trainer Simulator

Reflectone, Inc. of Tampa, Florida builds the A-10A Operational Flight Trainer for the Air Force through an initial 1976 contract with the Simulator Systems Program Office of the Aeronautical Systems Division. The principal mission of this trainer is to provide the capability of procedure

* Lead Flight Technology Engineer, Simulator System Program Office

** Grp. Ldr., Aerodynamics and Propulsion

+ Grp. Ldr., Flight Control Systems

and proficiency training of pilots required to fly the A-10A aircraft in fulfillment of its mission to navigate, seek out and destroy air and ground targets. The trainer provides the means of developing proficiency in all phases of instrument flight, including ground operations, takeoff, enroute navigation, holding, penetration, approach and landing under normal and emergency conditions. The visual system permits practice in normal and emergency procedures under simulated night visual conditions. Experience will also be gained in selection and release procedures associated with the basic armament system and simulated electronic warfare (EW) equipment.

Trainer Hardware

The trainer hardware and floor layout are shown in Figure 2. The primary computer capability consists of: three Scientific Electronics Laboratory (SEL) 32/55 computers (Units 1), an MDEC Vital IV system consisting of a Varian Image Generator Processor and Visual Display Unit (Units 7 and 2), and an LSI-11 minicomputer, aft of the cockpit station, for control loading modeling (Unit 8). Additional hardware systems are the Instructor station (Unit 3), the Electronic Warfare simulation cabinet and console (Unit 4), Audio cabinet system (Unit 5), and hydraulic and electrical power equipment (Unit 6).

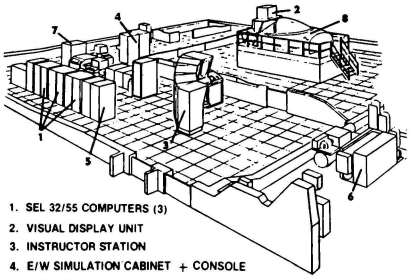


Fig. 2 A-10A Trainer Simulator Facility

Computers. Each SEL computer has 128K words of core memory of 16/32 bit words and seven input/output (I/O) channels. Communication between the three computers is established through the use of a memory interface adapter (MIA) creating a standard distributed processing system. One of the SEL computers is dedicated to electronic warfare modeling, another contains the majority of the aerodynamic and trainer simulation program modules, and the third is a master computer containing the executive, display, propulsion and navigation modules. Communication between the SEL computers and the visual system is conducted through a high speed data channel. The Varian visual computer has 32K memory with 16 bit words and produces a night only single visual scene for the pilot. A single CRT display is provided through the cockpit window. An identical display is provided via a color monitor at the instructor station. A bombing and strafing range along with five airport scenes are currently available from the visual system. The LSI-11 control loading minicomputer has 12K memory with 16 bit words and performs the flight control system modeling.

Cockpit System. The cockpit consists of an accurate replica of the A-10A serial #75-00262 aircraft. All instruments, indicators, panels, controls, and furnishings simulate the operational aircraft equivalent. The trainer does not have platform motion but does provide motion cues from limited seat pan, seat bladder, lap belt, and G-suit systems which are hydraulically and pneumatically driven. The G-seat consists of a three position hydraulically activated seat pan with a six cell bladder mounted in the active seat pan. The seat pan has three degrees of motion to a maximum of 2.5 inches. The back seat consists of a single position hydraulically activated back pan with a two-cell air bladder mounted in the lower back area. The pilot's seat and support are mounted on a vibration and buffet platform that provides two axes translation of one inch. Vibration cues are also provided through the control stick and rudder pedals.

Instructor Station. This area contains three Graphics 7 CRT terminals to present pertinent information for trainer initialization, modification, monitoring, and evaluation during training exercises. Pushbutton control of trainer vibration, communication, lighting, and G-suit pressurization are also at the instructor's console. Electronic warfare control and monitoring of procedures and proficiency via a CRT display are also available at the instructor station.

Trainer Software

The software is primarily coded in Fortran with some assembly coding primarily for the non-real-time programs. The trainer software system contains approximately 300,000 lines of code of which all are disc stored. Approximately 1,300 software (S/W) modules are employed in the following organization:

(1) Maintenance Programs
(2) Operating System
(3) Utility Programs
(4) Trainer Utility Programs
(5) Trainer Simulation Programs
(6) Trainer Data Libraries

The Maintenance programs contain software modules with built-in test functions which check hardware operation and interface signals. The Operating System programs contain the executive structure, real time monitor capability, and associated executive disc I/O output to the periferal handlers. These programs are all in assembly languages. The Utility and Trainer Utility programs contain support software consisting of linkers and loaders, compilers, assemblers and editors. The Trainer Simulation programs are those modules which represent the performance of the aircraft (e.g., the flight controls, aerodynamics, and propulsion systems). Thirty five percent of the total

trainer modules are Trainer simulation programs. The Trainer Data Library programs provide a common and shared base for software neumonics and data (e.g., CRT pages, EW mission environment, weather, initial conditions, record and playback data).

Flight Control System

The flight control system for the A-10 trainer simulator is a digital control loading system (DCL) designed and built in-house by Reflectone, Inc. The DCL represents a novel method for simulating the control systems of an aircraft. The traditional method has been a system consisting of sensors, mechanics, hydraulics, drive electronics, and modeling tied together with some I/O interface to some host CPU. In this design, the modeling electronics (basically an analog computer) was replaced with a digital computer. The control system modeling of the aircraft is now performed by the digital computer software. I/O interchange to the host SEL computer is now performed by direct memory address (DMA) exchanges between the LSI-11 digital control loader computer and the host CPU.

Control Loading Development. Design studies began in 1977 and revealed two basic conclusions. The first conclusion was that iteration rates relating control stick force, velocity, and position of at least 125 Hz were required. This was determined by placing a sample and hold circuit on an analog test fixture model of the aircraft flight control system and noting the sampling frequency at which quality began to degrade. The other basic conclusion was that rectangular integration proved most efficient. This was determined by Fortran runs of the control loading model. During this second study, numerical instabilities due to friction and spring pre-load discontinuities were discovered. The chronology of hardware and software development extending over a period of two years is shown in Table One.

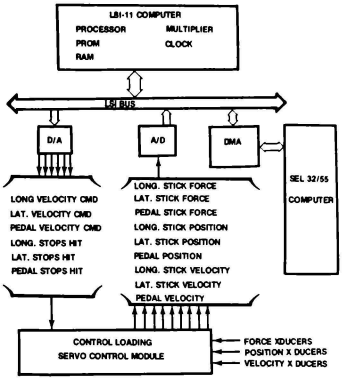


Fig. 3 Digital Control Loading Subsystem

Control Loading Function and Interface. The LSI-11 has fifteen (15) analog channels of which six are used as output and nine as input (See Fig. 3). Three of the six outputs are pitch and roll control stick, and rudder pedal velocity commands. The analog inputs to the LSI-11 are encoded and consist of three velocity, three position, and three force sensor inputs from the three axes of pitch roll, and yaw. All I/O signals between the LSI-11 and the control loading servo module as well as the LSI-11 computations are performed at 156 Hz. The LSI-11 communications with the SEL host computer through direct memory access (DMA) carrying 75 separate variables; the SEL sampling rate of these signals is at 20 Hz. The computational programs within the LSI-11 model each axis of control as two lumped mass bodies; a forward mass consisting

Table 1 - Control Loading Development

Design Change	Impact/Function
Designed control loading servo control module	Provided linkage between LSI-11 Computer and electromechanical servoactuators
Fortran program change	Added position stops and stabilization techniques for friction and preload nonlinearities
Designed receiver drive board	Provided interface electronics between LSI-11 and SEL computers
Developed roll tab shifter model	Unique aspect of A-10A flight control system capability
Introduced velocity control system	Improved system stability
Reintroduced H/W stabilization circuit	Force feedback into the servo valve drive amplifier
Implemented dual force scaling	Improved resolution and reduced noise
Introduced auto fine tuning	Improved overall system quality
Introduced 500 Hz dither signal	Reduced hysteresis and threshold
Introduced statistical fine tuning	Reduced scattering of fine tuning valves

of the control stick, pedals and bobweight, and an aft mass consisting of bellcranks, linkage, actuators, and the force feel system. Control system force, velocity and position sensors, along with mass properties, gearing, cable stretch, and nonlinearities describing the aircraft system, allow summation of forces and integration into a control position and velocity command signals.

Propulsion Module

The propulsion module consists of approximately 600 lines of Fortran code and is executed at 10 Hz with rectangular integration (See Fig. 4). Its primary outputs are Net thrust and Ram drag to the aerodynamics module, fuel flow to the fuel systems module, and thrust and engine speed to an aural cue module.

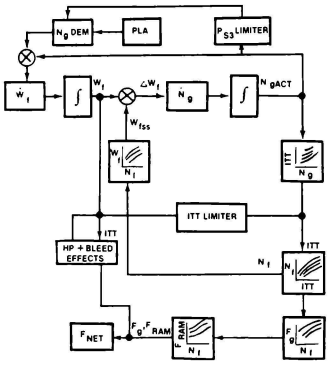


Fig. 4 Propulsion Module

Operation. Throttle positions (PLA) command a gas generator core RPM (N_gDEM) limited by compressor discharge pressure (P_{S3}). This signal is differenced with actual core RPM (N_{gACT}) resulting in a demanded fuel flow rate change ($\dot{W}_f$). The integral of this signal is compared with computed steady state fuel flow (w_{fss}) commanding an actual gas generator RPM rate of change (Ng). This signal is integrated into actual core RPM which, through data tables as a secondary function of altitude, determine interstage turbine temperatures (ITT). Interstage turbine temperature is used to compute fan speed (N_f) and to limit commanded fuel flow. Gross thrust and Ram drag are computed using fan speed and Net thrust is finally obtained by taking the difference between Gross thrust and Engine ram drag.

Module Interface. The engine module interfaces with seven other software modules and the cockpit systems (See Fig. 5). The engine module operates a number of engine status lights and indicators in the cockpit. Signals for throttle position, ignition, fuel flow and fire handles are received from the cockpit through discrete (DI) and continuous signals (AI).

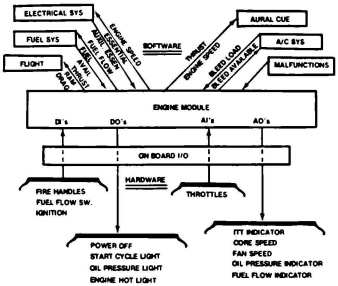


Fig. 5 Propulsion Hardware/Software Interface

Flight Module

The flight module, FFLITE, consists of approximately 550 lines of code and performs the calculations and summation of all aerodynamic forces and moments acting at the aircraft center-of-gravity. These force and moments are function of configuration, control surfaces, engine thrust, angle of attack, sideslip, body angular rates, and environmental conditions (i.e., Mach No. and altitude). The module data is of approximately 60 percent tabular and 40 percent equation form. The data base is a mix of wind tunnel and flight test data obtained from the Air Force Flight Test Center at Edwards Air Force Base, California, and Fairchild Republic Aircraft Company.

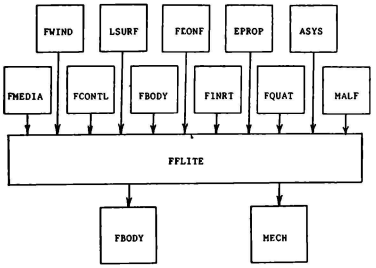
Description

The basic set of aerodynamic data and equations are based upon a cruise configured A-10A with a center of gravity corresponding to the 25 percent mean aerodynamic chord of the wing. Force and moment changes due to configuration changes i.e., flaps, speedbrakes, and external stores are generally added as incremental effects to the basic cruise configured tabular aerodynamic data set. Aerodynamic effects due to buffet, ground effects, and asymmetric control surface deflections are also modeled. Wing stall conditions are also determined by computing local wing angles of attack based on fuselage angle of attack and rotational body rates. Incremental post stall forces and moments are computed as a consequence of wing tip stall conditions. Side force due to rudder and lift force due to elevator are modeled as geometric multipliers of yawing moment due to rudder and pitching moment due to elevator respectively. Reductions in lift and a nose down pitching moment due to the absolute value of sideslip are also included. These values are 15 and 10 counts of lift coefficient and pitching moment respectively per degree of sideslip.

Interface

The flight module receives input from eleven other systems and outputs data to basically two modules (See Fig. 6). Most of the modules in Figure 6 execute at 20 Hz. The total aerodynamic

force and moment outputs of the flight module are summed with thrust and gravity forces then integrated to body axes angular rates and velocities within the FBODY module which contains the six degree of freedom body axes equations of motion. The mechanoreceptor (MECH) module uses the flight module outputs to compute hydraulic and pneumatic drive signals for the cockpit G-seat and anti-G-suit.



FMEDIA-Environment
FWIND-Wind Earth Velocity
FCONTL-Primary Controls
LSURF-Secondary Controls
FCONF-Weapons
FINRT-Weight & Balance
EPROP-Propulsion
FQUAT-Quaternions
A SYS-Systems
MALF-Malfunctions
MECH-Mech-receptor
FBODY-Equations of Motion

Fig. 6 Aerodynamic Module (FFLITE) Interface

Acceptance Test Problems and Corrections

A number of flight system performance differences were observed between the trainer simulator and the A-10A aircraft during trainer testing. The more interesting of these are presented and discussed.

Attitude Changes

Slow yaw, pitch, and roll control attitude changes were observed on instruments and the visual system following a controls free, careful one-G setup. The motion drift was caused by fluxuating surface control positions from the LSI-11 digital control loader into FFLITE, the aerodynamics module, causing subsequent attitude changes. It was primarily solved by creating a S/W statistical automatic fine tuning loop for the flight control system in pitch, roll, and yaw to account for null effects. This logic looks at stick and pedal velocity, and force over 64 sampling periods (less than one second at a sampling rate of 156 Hz.) then statistically averages the values and uses them as new and current bias and null forces for servovalve commands. A secondary fix involved a separate hold (clamp) circuit for the control surfaces to eliminate small signal disturbances. Stick position and velocity, surface control and trim positions are used in the hold circuit logic. When these position and velocity signals are within a small magnitude level, a new current trim position is used.

Tracking/Strafing

The pilot control workload associated with capturing and maintaining the Head-Up-Display (HUD) gun cross on the target during acquisition and tracking was excessive by comparison with the aircraft work load required. This was an early recognized deficiency of the simulator which was also one of the last problems corrected; it was solved by deviating from design yaw stability augmentation criteria. The design criteria called for an aileron to rudder interconnect (ARI) gain of .2 degrees of rudder per degree of aileron below 240 knots to 0.4 degrees per degree above 240 knots along with a yaw rate wash out circuit of four (4) seconds, (4s/4s+1), degrees rudder per degree/second of yaw rate. This SEL computer circuit was modified, during pilot evaluation, to a fixed value at all airspeeds of .8 degrees/degree for the ARI with a 16 second washout, (16s/16s+1), for the yaw rate damper. This change not only resulted in clearing this discrepancy but also another characterized by excessive yaw and roll during final landing approach. This change appears to be a situation where some undetected (negative) characteristic of the simulator was corrected by an intentional design deviation (negative) resulting in a (positive) acceptance performance result. Prior to making this change the visual brightness of the gun strafing target was poor and was brightened. This visual change reduced pilot workload but not of a sufficient nature to preclude this yaw SAS change.

Stick Motion

There was perceptible and disturbing jerkiness and slow start up and stop in the control stick position in the pitch axis during continuous and transient operation of the normal and emergency trim circuits. This was solved by adding a first order lead circuit to the commanded stick position from the LSI-11. A small force bias was also added to the circuit. The basic computations for the stick position signal due to trim occurs in the SEL computer which contains a first order lag circuit computed at 20 Hz. A 500 Hz dither signal was also added to the servovalve to keep the spool moving at all times and reduce friction; it, however, had minimum impact on the solution of this problem.

Trim Workload

Excessive pilot workload was necessary to maintain a one-g trim flight condition. The cockpit vertical velocity indicator (VVI) was recognized as the main problem when it was covered over and the pilot cued off the horizon from the visual scene and was then able to trim the simulator much quicker and easier. It was apparent from this effort that the VVI rate of climb signal was too fast so its lag was increased from a value of .3 to a value of .15. These values were multiplying factors applied to the instantaneous difference between computed and displayed values of instantaneous vertical velocity. This lag effect reflects the instrument display lag between the instantaneous and displayed vertical velocity as a consequence of the aneroid hole around the VVI bellows.

Stall Speeds

The simulator stall speeds (maximum lift coefficients) and stall angles of attack did not vary adequately with power, configuration, store loadings, or Mach number using the wind tunnel aerodynamic data while testing against flight data.[1] A data base update was performed to the lift and angle of attack characteristic of the AFFTC flight data reports. A certain amount of care was necessary in order to use the flight test data. It provided lift versus angle of attack at low Mach number, and maximum lift coefficient values at higher Mach numbers as functions of power setting, flap and speedbrake. This data excluded geometric lift due to thrust but included elevator trim lift corresponding to a 25 percent m.a.c. center of gravity. This flight data information was used to restructure the programs lift equations and accurately represent the various maximum lift coefficient and stall angles of attack obtainable from the aircraft.

Manual Reversion Transients

The pitch rate, and normal acceleration transients associated with the transfer of control from normal (powered) flight controls to manual (unpowered) controls was unrealistic-particularly at forward center-of-gravity conditions where the simulator pitched up rather than down (like the aircraft). The direction and attitude response of the simulator associated with this control transition is affected by lift and moment changes caused by the time and degree of symmetric upfloat of the aileron, and float of the elevator to zero hinge moment conditions. One portion of the problem fix was an increase in flexibility of the control system cabling (a spring constant reduction) resulting in increased control surface deflection for a given airload at a fixed control position.

Instrument Ball Motion

The attitude direction indicator ball motion during small heading and attitude changes was excessive and disconcerting to the pilot and impacted his perception of straight flight and controllability. This was corrected by increasing the damping of the side force acceleration computation to the ADI from a value of 0.75 to 1.5.

Stall Pitchout

Pitch characteristics at stall were unsatisfactory varying during development from a strong pitch down to-neutral pitch to-an acceptable and representative mild pitch down. Velocity bleedoff above stall angles of attack was also excessive. These characteristics were modified and corrected by nonlinear changes made to lift and drag characteristics at and beyond stall angles of attack. Changes to static pitch stability with angle of attack, C_{m_α}, and elevator sensitivity were not necessary. Post-stall drag polars were modified from a first order extrapolation with lift to a second order algorithm.

Take-Off Rotation Speeds

Two problems existed which, when recognized as being related, were both corrected with a single design change. The first was the minimum rotation speed at takeoff which was higher than the aircraft. The other problem was an excessive pitch down with a throttle chop (the aircraft has negligible initial pitch change with throttle motion). A review of the software model engine line-of-thrust and design data showed that the pitching moment associated with tailpipe curvature (4.5 degrees up) had not been accounted for. Inclusion of this effect in the model solved both the higher rotation speed and the initial pitch up with a throttle chop. An added benefit was the readjustment of C_{m_o} values back to magnitudes published in flight test reports. Prior to recognition of the tailpipe curvature effect the flight values of C_{m_o} had been considerably modified to correct trim elevator differences between the simulator and flight results.

Climb Performance

Climb performance was excessive at higher attitude. This was corrected by refinements to the engine limiter logic. At low altitude, the engine is primarily limited by interstage turbine temperature, gas generator speed, and discharge pressure whereas at higher attitudes the interstage turbine temperature is the primary engine limiter function. A proper blending of these limiters resulted in a correction to the excessive high altitude rate of climb problem.

References

1. McLaughlin, M.J., et al, A-10A Thunderbolt II Performance Evaluation, Air Force Flight Test Center TR-78-2, June 1978.

2. Air Force Flight Test Center Report TR-77-11, A-10A Flying Qualities Air Force Developmental Test and Evaluation, Sept, 1977.

3. Mathematical Model Reports, Computer Product Specification, Reflectone, Inc., 1979.

S. Ramachandran
Goodyear Aerospace Corporation
Akron, Ohio 44315

R.T. Galloway
Air Warfare Systems Division
Naval Training Equipment Center
Orlando, Florida 32813

Abstract

A methodology to provide spin training for the TA-4J pilots was formulated and implemented on the TA-4J Operational Flight Trainer (Device 2F90). User experience indicates the simulation provides worthwhile training in erect spin recognition and recovery procedures. This paper presents details of: formulation of simulation requirements, development of spin simulation, acceptance test methodology and user experience.

Introduction

Pilot training in out-of-control flight enhances the flying skills of the pilot, enabling him to exploit the full capability of the airplane with increased confidence and safety. However, actual practice of departures from controlled flight in some U.S. Navy tactical aircraft can lead to dangerous situations which must be tempered by flight restrictions to avoid loss of the aircraft. Such restrictions apply to the TA-4J airplane with respect to intentional spins. The TA-4J is utilized by the U.S. Naval Air Training Command as an advanced jet trainer where a part of the syllabus includes aerobatics and air combat maneuvering. Intentional spinning is not permitted in this airplane but occasionally spins do develop from pilot error during out-of-control flight situations. In fact, loss of TA-4J aircraft in stall/spin incidents occurs on a continued basis during maneuvering flight training. In an effort to reduce these losses, the Chief, Naval Air Training (CNATRA) searched for a means to augment the existing pilot training aids for TA-4J spin characteristics, namely the narrative discussion contained in the NATOPS Flight Manual and a training film developed from spin flight tests. Attention was directed toward the TA-4J flight simulator, known as Device 2F90 Operational Flight Trainer (OFT), to provide hands-on pilot training in spins, but tests demonstrated that the existing data base did not accurately simulate TA-4J spin characteristics. Therefore, CNATRA established the requirement to add a spin training capability to the OFT, and the U.S. Naval Training Equipment Center (NAVTRAEQUIPCEN) and Goodyear Aerospace Corporation undertook the development of the required modification. This was a unique effort in that prior to this, a Navy flight simulator has not been specifically modified to provide spin training. This paper presents details of: the formulation of simulation requirements through analysis of the flight test data available; the development of the spin simulation to meet the simulation requirements; and finally acceptance test methodology and user experience.

Description of the TA-4J Airplane

The TA-4J airplane (Figure 1.) is a two place (tandem) trainer version of the A-4 series attack airplane designed and manufactured by the McDonnell-Douglas Corporation for carrier and land based operations. Its identifying features include a low wing with a modified delta planform, swept back empennage, and a large canopy area. The spin characteristics of the TA-4J airplane have been tested in a series of flight test programs conducted by the airframe contractor and the U.S. Naval Air Test Center (NAVAIRTESTCEN). Test results are documented in References 1 and 2.

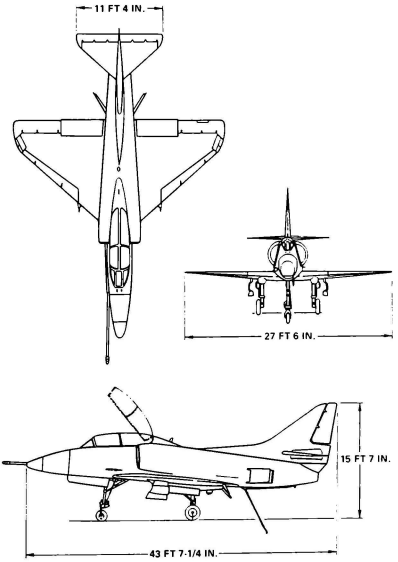


FIGURE 1. TA-4J AIRCRAFT

The Reference 1 effort placed primary emphasis on evaluating the suitability of the TA-4J (then designated TA-4F) airplane as a spin trainer. Erect and inverted spins were evaluated with

various wing store loadings and from a variety of entry conditions. Within the scope of these tests, the TA-4J was spin resistant and exhibited no tendency to spin inadvertently. Erect spin modes were either the highly oscillatory diving spiral mode or the classic erect spin mode, depending on external wing store loading. Inverted spins were attainable in all loadings tested. The erect spin modes were not disorienting and caused no pilot confusion, but inverted spins were usually violently oscillatory, disorienting, and could be structurally damaging due to negative g overstresses.

Reference 1 recommended that intentional erect spins be permitted in the TA-4J airplane since they were sufficiently predictable. Intentional inverted spins were not recommended because of structural and recovery problems. However, it was decided not to permit any intentional spins primarily because incorrect pilot technique with lateral control could result in a rapid transition from an erect to an inverted spin.

Description of the TA-4J OFT

The TA-4J OFT (Device 2F90), (Figure 2), built by Goodyear Aerospace Corporation, is a three degree of freedom moving base simulator. Each unit or deck consists of a set of four cockpits and instructor consoles all operated by a pair of Xerox Sigma 5 digital computers and a common hydraulic power supply system. Each cockpit interior is identical to the forward cockpit of the TA-4J airplane with respect to internal measurements, cockpit instruments and controls. Each instructor console consists of instruments and controls for monitoring and modifying flight conditions to simulate various emergencies and malfunctions. The motion base provides limited motion in pitch ($\pm$15 degrees), roll ($\pm$15 degrees) and heave ($\pm$6 in.) and is capable of simulating buffet and turbulence.

A visual system is attached to one cockpit at each of three training sites. This visual system, built by General Electric Corporation, provides low resolution, color, computer generated images (CGI) displayed on three large screens placed in front of the cockpit. These screens provide a field of view consisting of $\pm$ 105 deg horizontally and $\pm$ 30 deg vertically.

The simulation of TA-4J flight characteristics provided by the OFT are considered quantitatively and qualitatively representative for pilot training in normal instrument, aerobatic, and emergency procedures tasks. The simulation of high angle of attack (AOA) characteristics had been validated only for stalls entered from erect (as opposed to inverted) flight. While the simulation did exhibit post stall gyrations when spin entries were attempted, these gyrations were not considered representative of the TA-4J airplane characteristics.

The Spin Simulation Problem

The airplane spin phenomenon, as defined in Reference 5, is an uncontrolled, large angle, six degree of freedom motion experienced by an airplane operating in the stalled aerodynamic region. The spin is a highly complex motion influenced by a host of non-linear variables including the aerodynamic aspects of the airframes, aircraft mass distribution, and maneuver entry flight conditions such as airspeed, normal acceleration and flight control deflections. The complexity of the spin motion is indicated by the variety of descriptors required to characterize each spin mode such as: erect or inverted, steep or flat; oscillatory or steady; and low or high rates of spin rotation or angular velocity.

From a simulation viewpoint, the airframe aerodynamic parameters above stall AOA are the most difficult variables to define. In addition, it is

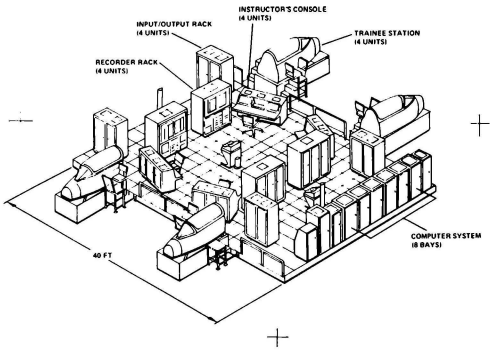


FIGURE 2. DEVICE 2F90

rare that a comprehensive set of actual test data exists for use as a criteria source for thorough validation of a spin simulation model. Thus, the designer of a spin simulation is faced with some large gaps in terms of input data (aerodynamic coefficients at high AOA) and output data (time histories of airplane response during various spin modes and with various control deflection combinations).

It is obvious that some amount of output data must be obtained before any simulation can proceed. These data help define mass distribution and entry conditions, but the simulation designer still must often develop his own aerodynamic coefficients for input data. A common approach is to apply multiplication factors to the classical aerodynamic derivatives. An iterative approach is then utilized to adjust each factor until, hopefully, all the output time history characteristics have been matched. However, since the effect of each term on spin characteristics is not clearly understood and the spin motion is very complex, such an iterative method is very inefficient without an extensive set of time histories and the services of test pilots and aerodynamicists knowledgeable in the particular aircraft's spin characteristics. Excellent simulations of departure characteristics have been developed in this manner in flight simulators for the A-7E and F-14A airplanes in Devices 2F111 and 2F95, respectively. However, without data and expert guidance, the iteration process may only produce a match of gross characteristics such as turn rate and altitude loss at the expense of other important characteristics such as attitude, oscillation phasing, and response to incorrect control inputs.

Another approach to aerodynamic modeling is the application of parameter identification techniques (Reference 6). These techniques have been gaining popularity in recent years as viable means of determining aerodynamic characteristics from flight data although they have not been extensively applied toward improving simulation fidelity. In order to be successful, these methods require a large amount of high quality flight test time histories as well as wind tunnel data. Lack of such a data base precluded the application of parameter identification techniques to this spin simulation.

A third concept for modeling spin aerodynamic derivatives involves consideration of rotary balance effects as discussed in Reference 7. These rotary balance effects arise from the fact that airflow impinges on opposite sides of the airframe simultaneously when the airplane is rotating about a vertical axis near the center of gravity. In Reference 7, rotary balance coefficients were shown to be effective in modeling spin characteristics and they could be determined directly in special wind tunnel tests. However, no rotary balance coefficient data were available for the TA-4J airplane.

The TA-4J spin simulation had to be accomplished with very limited flight test data and test pilot expertise. The only TA-4J data readily available were: a limited set of time histories published in Reference 1; the narratives in the same report and in the NATOPS Flight Manual; (Reference 4), and the TA-4J spin training film (Reference 3). Other flight test data existed in unpublished form from follow-on programs, but these data provided no amplification over information already at hand. A further problem was that there were no Navy test pilots available with current spin experience in the TA-4J airplane since the last set of official spin tests had been concluded several years ago. Test pilots with relevant spin experience in other aircraft types were available, but they were considered useful for final acceptance testing only, since they did not have sufficient experience to assist in development. Therefore, this spin simulation had to be developed from a very limited data base, and then validated by engineers and pilots who had no first-hand experience with TA-4J spins. This meant that the design and test requirements had to be as explicit as possible to ensure efficient development and validation of the simulation.

Requirements Formulation

The original request from fleet instructor pilots for spin simulation in the TA-4J OFT expressed a need for training capability in both the erect and inverted spin modes. These pilots envisioned a capability in the simulator for re-enacting stall/spin accidents that occurred inadvertently due to pilot error in maneuvering flight. However, the NAVTRAEQUIPCEN team felt that there was too little data available to achieve this capability. In addition, it was felt that training for a very dynamic situation such as spins should be structured as much as possible to insure that the student pilot could clearly identify cause and effect relationships due to both proper and improper cockpit control actions. Therefore, the spin simulation requirement had to describe a complex phenomenon in straight forward terminology in order to insure that training objectives could be met.

The spin simulation objectives were developed through discussions between CNATRA and NAVTRAEQUIPCEN. The primary objective was to provide training in proper cue recognition and psychomotor responses to enable safe recovery from out-of-control flight. This objective had to be balanced against several limiting factors, the most significant of which were limited availability of criteria data and test pilot expertise, limited computer memory capacity in the OFT and a relatively modest funding level. It was decided to model the training capability of the OFT after the recommendations of Reference 1, which were developed for spin training in the actual airplane. Since intentional inverted spins were not recommended in Reference 1, the decision was made to simulate the erect spin modes only, and to exclude simulation of the inverted spin characteristics. The decision to eliminate the complex inverted spin modes alleviated concern over achieving some measure of success within the constraints mentioned above.

Once the scope of the spin simulation had been confined to the erect modes, the training objectives were further refined to include the effects of any combination of cockpit control inputs and to insure that spin characteristics were accurately displayed on the cockpit instruments, especially

those of primary importance to the pilot in a spin situation, namely, angle of attack, turn needle, attitude indicator, and airspeed indicator. The effects of peripheral cues derived in the simulation from the motion and visual systems were considered secondary due to the limited capabilities of these systems, and these cues were to be included in the model only to the extent that they did not induce negative aspects to the training. In summary, the objectives of the spin simulation were to provide proper response characteristics for the erect spin modes of the TA-4J airplane as seen on the cockpit instruments. All combinations of cockpit control deflections were to be considered in order to demonstrate the effects of both proper and improper pilot control actions.

The desired spin characteristics were described in a tabular format as described in the Appendix which also includes an explanation of the test techniques and nomenclature. Table A2 of the Appendix outlines the erect spin mode tests and Table A3 of the Appendix outlines the diving spiral mode tests. The contents of the Appendix were developed at the NAVAIRTESTCEN by careful analysis of the airplane spin documentation. It was fortunate that both the NAVAIRTESTCEN flight test report (Reference 1) and the spin training film (Reference 3) contained thorough narrative discussions of significant spin characteristics upon which to base the expected response during the entry, steady state, and recovery phases. In addition, Reference 1 contained detailed discussions of the effects of misapplied controls and other effects (such as speed brake and power settings) which were used to formulate these expected response requirements. The referenced narratives were thorough enough so that reliance on the relatively crude time history traces in Reference 1 for detailed analysis was minimized. In addition, commonality of the simulation with existing fleet spin training aids for the TA-4J airplane was ensured by utilizing the same narrative material.

Simulation Method

The simulation technique employed here has several advantages. It is general in nature and can be easily adapted to other flight simulators. The computer requirements are minimal - the TA-4J erect spin simulation uses only 1550 words of computer memory. It is an effective method to train the pilots in the basic procedures of out-of-control flight without an extensive aerodynamic data base or the services of experienced pilots and aerodynamicists.

The underlying philosophy is to provide the necessary cues to the pilot for a successful spin training using simple equations that directly drive the instruments and motion and visual systems and replicate the basic spin characteristics outlined in Tables A2 and A3 without satisfying the mathematical constraints of the equations of motion. The TA-4J exhibits consistent tendencies in all phases of spin, departure through recovery, even though the actual values of variables such as angle of attack may differ considerably from one flight to another. The simulation method takes advantage of this predictability.

The erect spin methodology is discussed below and the diving spiral simulation is similar. The aircraft has three definitive phases, namely entry/incipient spin, fully developed spin, and recovery, details of which have been given in the Appendix. Recovery is effected by any of the right combination of control inputs and the recovery rate depends on the applied control input. Premature removal of recovery control will result in the aircraft reverting to spin. Based on the flight condition, the phase that is presently simulated, and the control inputs applied, switching is done from one phase to another while maintaining a smooth transition. An example is given below for the roll angle and other variables are modeled in a similar fashion. To avoid repetitiveness, wherever possible, the magnitudes of variables such as recovery attitude, time period of oscillation, etc., are chosen randomly for each simulation. Once the turn rate goes to zero, the simulation is switched back to the aerodynamic model and the pilot flies out of the dive with the aerodynamic simulation.

1. Roll angle (ϕ) during incipient erect spin:

(a) $t_o < t < t^*$

$$\phi_{i+1} = \phi_i + \dot{\phi}\,\Delta t \tag{1}$$

$$\dot{\phi} = \frac{360\ KSIGN - \phi_o}{t^* - t_o} \tag{2}$$

where t_o is the time at which departure occurs, t^* is the time at which one roll is completed, t is the time, ϕ_i and ϕ_{i+1} are current and next values of ϕ respectively, ϕ_o is initial roll attitude in degrees, and Δt is computer frame time. KSIGN is ± 1 and is equal to sign $(-\delta_{r_o})$ to ensure roll in the applied direction of the rudder. δ_{r_o} is the rudder deflection at t_o.

(b) $t > t^*$

$$\phi_{i+1} = \phi_{a_{i+1}} \left[\sin \frac{2\pi (t_{i+1} - t^*)}{T_\phi} \right] \cdot KSIGN \tag{3}$$

$$\phi_{a_{i+1}} = \phi_{a_i} + (\phi_{a_c} - \phi_{a_i}) \frac{\Delta t}{T_\phi} \tag{4}$$

where T_o is the period of oscillation for toll angle and ϕ_{a_c} is the desired amplitude of oscillation. The variable ϕ_{a_i} is initialized at t^* to the value of roll angle at t^*.

Equation (4) allows the magnitude of oscillation to reach the desired value in an exponential fashion.

2. Roll angle during steady erect spin:

$$\phi_{i+1} = \phi_{a_{i+1}} \left[\text{Sin} \frac{2\pi (t_{i+1} - t^*)}{T_\phi} \right] \text{KSIGN} \quad (5)$$

$$\phi_{a_{i+1}} = \phi_{a_i} + (\phi_{a_c} - \phi_{a_i}) \frac{\Delta t}{T_\phi} \quad (6)$$

$$\phi'_{a_c} = 3|\delta_a|, \text{ if } |\delta_a| > 1/3\ \delta_{a_{max}}$$

$$= 20, \text{ otherwise}$$

$$\phi''_{a_c} = \Delta\phi; \text{ if } \delta_e > a_1$$

$$= 0, \text{ otherwise}$$

$$\phi_{a_c} = \phi'_{a_c} + \phi''_{a_c}$$

where $_a$ is the aileron deflection, $\delta_{a_{max}}$ is the maximum aileron deflection, δ_e is the elevator deflection, $\Delta\phi$ is increase in roll due to forward elevator and a_1 is a constant, all in degrees; other variables are as defined earlier. Note that the above formulation permits a smooth change in the roll angle oscillation under all conditions.

3. Roll angle during recovery from erect spin:

The number of turns to recovery depends on the recovery control as shown in Table A2. The flight test data indicated that the attenuation of spin is gradual in the beginning but the turn rate rapidly goes to zero in the last 1/4 turn (when turn rate reaches about 40 degrees/sec). Hence, the recovery simulation has been divided into two phases to reflect this. The equations for ϕ are given below.

(a) Turn rate $\geq$ 40 degrees/second:

$$\phi_{a_{i+1}} = \phi_{a_i} - \frac{\phi_{a_i} \Delta t'}{T_r + T_f} \quad (7)$$

$$\phi_{i+1} = \phi_{a_{i+1}} \text{ Sin } \frac{2\pi (t_{i+1} - t^*)}{T_\phi} \text{ KSIGN} \quad (8)$$

where $\Delta t' = \Delta t \cdot \frac{N_{opt}}{N}$

In the above equations, the decay rate of ϕ is adjusted through $\Delta t'$ to reflect the type of recovery control. N_{opt} and N are the number of turns for optimum recovery control and the actual recovery control applied, respectively. T_r is the time required to reach a turn rate of 40 degrees/sec had optimum recovery been applied during the existing flight conditions and T_f is the time it will take the spin to completely stop from a turn rate of 40 deg/sec.

T_r and T_f are calculated as part of the simulation

(b) Turn rate $<$ 40 degrees/second

$$\phi_{i+1} = \text{KSIGN } \phi_{a_{i+1}} \text{ Sin } \frac{2\,(t - t^*)}{T_0} \quad (9)$$

where,

$$\phi_{a_{i+1}} = \phi_{a_i} - \frac{\phi_{a_i} \Delta t}{T_f - t + t'} \quad (10)$$

In the above equation t' is the time at which the turn rate reaches a magnitude of 40 degrees/second. $(T_f - t + t')$ goes to zero only when the turn rate becomes zero when the spin simulation is no longer required.

Figure 3a shows a classic erect spin time history recorded in flight. Figure 3b shows a similar case flown in the simulator. It may be noted that the recovery with optimum controls occurs within 1½ to 3 turns in both cases and the trends on bank angle and heading are similar. The angle of attack for the simulator run is the cockpit instrument reading and hence is pegged at 30 units as per Table A2 whereas the nose boom angle of attack reading was recorded in flight tests.

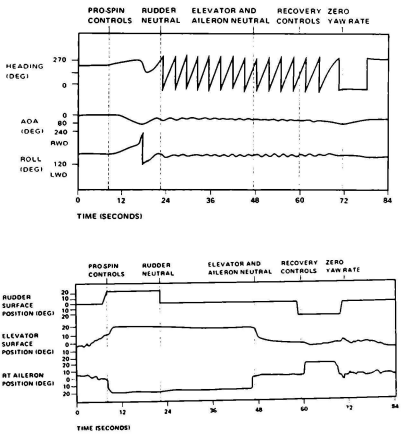


FIGURE 3a. CLASSIC ERECT SPIN (FLIGHT TEST)

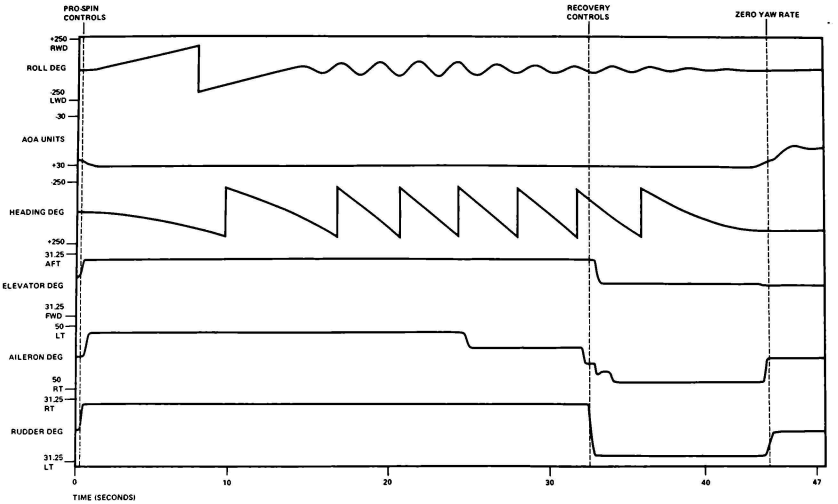


FIGURE 3b. CLASSIC ERECT SPIN (SIMULATOR)

Testing the Simulation

The acceptance tests for the spin simulation at NAS, Meridian, MS consisted of pilot evaluation by local instructor pilots and a spin experienced test pilot from NAVAIRTESTCEN, and formal engineering test procedures based on the tests in the Appendix, which included time history recordings of significant spin parameters. At the conclusion of these tests, the simulation was considered sufficiently representative of TA-4J spin characteristics for use in training student Naval aviators.

The pilot evaluation portion of this program was somewhat unusual in that none of the pilots had actually experienced spins in the TA-4J aircraft. However, the test pilot from NAVAIRTESTCEN was serving as the Staff Spin Instructor Pilot for the U.S. Naval Test Pilot School and he was very knowledgeable in typical spin characteristics. He had regularly demonstrated spin characteristics to student test pilots and fleet fighter pilots using the T-2C airplane in programs such as described in Reference 8 and his spin experience included several types of jet aircraft, some of which were very similar to the TA-4J. This pilot evaluated the simulated spin characteristics using the test techniques of the Appendix and other entry and recovery methods. Several minor deficiencies were identified which were subsequently corrected in the software, but in general, the test pilots considered the simulation representative of this type of aircraft except for certain limitations imposed by design, i.e., maneuver predictability and minimal visual cues. Maneuver predictability concerned the fact that while the airplane steady state spin characteristics were predictable, the stall and post stall gyration preceding spin entry were very dependent upon entry conditions, particularly airspeed and normal acceleration. The simulation is inherently predictable in these areas and it was recommended that the student Naval aviator not enter a spin in the simulator in other than one g, wings level flight. The visual cues available in the OFT cockpits so equipped are minimal during spins due to the limited field of view. It was recommended that student Naval aviators rely primarily on cockpit instruments in the simulator to determine spin characteristics and proper recovery controls. This recommendation coincided with the original simulation design intent to teach recognition of spin characteristics based on cockpit instruments since it was well known that pilots could be mislead and become disoriented when trying to reference the outside horizon during spins in the airplane.

The fleet instructor pilots were asked to conduct each of the test maneuvers in the Appendix to become familiar with the simulation capabilities. They were then asked to note any anamolies with respect to flight training requirements, their flying experience, and their knowledge of published information on TA-4J spin characteristics (References 1 and 4). In addition, the instructor pilots observed the tests conducted by the NAVAIRTESTCEN

test pilot. The primary complaints expressed by these pilots were the lack of inverted spin capability and the restriction of spin entry to wings level, unaccelerated flight but they agreed to introduce the spin simulation into the OFT syllabus and to observe its effectiveness before recommending further development or corrective action.

User Experience

The spin simulation was initially implemented in the syllabus at NAS, Meridian as part of the last visual-equipped simulator "flight" for the student Naval aviator prior to flying the TA-4J airplane. After several months of utilization, the instructor pilots now feel that the spin simulation provides worthwhile training in the procedures required to recognize and recover from erect spins. The repeatability aspect of the simulation is considered an asset because it enhances the instructor's credibility in taking a student pilot through the spin demonstration. The simulation has also been educational for the instructor pilots themselves because they did not always apply the right controls for recovery in their first simulator practice sessions. Training value is considered somewhat limited because the violent disorienting motion cues that are characteristic of spins could not be provided due to OFT motion system limitations. The lack of inverted spin characteristics in the simulation remains a serious concern

Conclusion

A successful simulation of the erect spin characteristics of the TA-4J airplane was developed and implemented to provide training for student Naval aviators. This effort is significant because of constraints imposed by limited availability of aircraft data and test pilot expertise, and by limited computer memory space available in the OFT. Certain limitations were imposed on the simulation by design to ensure meaningful instruction capability within program constraints. The most significant design limitations were the exclusion of the inverted spin modes and restriction of erect spin entry to wings level, unaccelerated flight conditions. The technique formulated for this spin simulation was general in nature and could be easily expanded or applied to other flight simulators. The spin simulation underwent testing conducted by engineers, by a spin-experienced test pilot from NAVAIRTESTCEN, and by fleet instructor pilots and it was considered representative of the TA-4J airplane erect spin characteristics. User experience indicates the simulation provides worthwhile training in erect spin recognition and recovery procedures. The lack of an inverted spin simulation remains a serious user concern but the instructors consider the inherent predictability of the model as an asset in demonstrating spin characteristics to student Naval aviators.

Acknowledgement

The authors wish to thank Otis Vestfals and Wayne P. Leavy of Goodyear Aerospace Corporation for their significant contributions to this program.

References

1. "Navy Spin Evaluation of the TA-4J Airplane," Naval Air Test Center Technical Report FT-8R-72, Feb 1972.

2. "Departure and Recovery Characteristics from Nose-High Low-Airspeed Conditions and High-Speed Dive and Dive Recovery Characteristics of the TA-4F/A-4M Airplanes," Naval Air Test Center Technical Report SA-22R-76, Apr 1976.

3. "TA-4F High Angle of Attack/Spin Flight Characteristics," USN Film No. MN10870.

4. "NATOPS Flight Manual for the Model TA-4F Aircraft," NAVAIR 01-40AVD-1, May 1978.

5. Bihrle, W. and Heyman, A.C., "The Spin Behavior of Aircraft," GAEC Report No. 394-68-1, Dec 1967.

6. Ramachandran, S., et al, "Identification of Aircraft Characteristics at High Angles of Attack and Sideslip Using the Estimation-Before-Modeling (EBM) Technique," Paper 77-1169, Proceedings of the AIAA Atmospheric Flight Mechanics Conference, Hollywood, FL, Aug. 1977.

7. Bihrle, W. and Barnhart, B., "Effects of Several Factors on Theoretical Predictions of Airplane Spin Characteristics," NASA CR-132521, Aug 1974.

8. North, D.M., "Navy Spin Training Heightens Confidence, Pilot Safety," Aviation Week and Space Technology, August 25, 1980.

APPENDIX:

TA-4J OFT
ERECT SPIN TEST METHOD AND TEST CONDITIONS

1. Erect spin characteristics consist of two modes: classic erect spin mode and diving spiral mode. Entry into each mode is a function of external fuel quantity and cockpit control position. Typical initial conditions are presented in Table A-1. The pilot technique for entering either spin mode is as follows: set longitudinal trim as required by test procedures while remaining stabilized at approximately 150 KIAS; decelerate at approximately 2 to 5 kt/sec by applying aft stick; when AOA reaches 20 to 21 units, smoothly apply full rudder input in the direction of the spin and slowly place the stick full aft followed by the desired lateral stick input.

2. The simulation will exhibit the appropriate post stall gyration followed by the steady state mode as outlined in Tables A2 and A3. Once the post stall gyration has commenced, any combination of control inputs can be attempted to observe the effect on spin charistics. (See Tables A2 and A3.) The symbology utilized to denote cockpit control positions in the spin test tables is shown in Figure A-1.

TABLE A-1 - INITIAL CONDITIONS

Altitude:	30,000 ft	Internal Fuel:	3,000 lb
Airspeed:	150 KIAS	Landing Gear and Flaps:	Up
Speedbrake:	In	Power:	Idle

Spin mode	Loading	External fuel quantity
Diving spiral	No stores or two 300 gal drop tanks	None
Classic erect spin	Two 300 gal drop tanks	More than 2000 lb

NOTE

Initial altitude, airspeed, internal fuel, power setting, or speedbrake position are not critical for obtaining either spin mode; the values presented are just suggested conditions based on airplane test results. The landing gear and flaps must be up to obtain either spin mode. If external fuel quantity is between zero and 2,000 lbs, then either spin may occur.

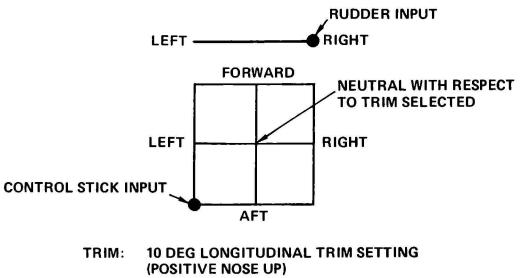


Figure A-1. Cockpit Control Position Symbology

TABLE A-2 - ERECT SPIN MODE TESTS

Test item	Maneuver phase	Control input	Expected response
1. Typical incipient spin (to the right)	Entry and incipient spin (controls applied at 20 to 21 units AOA)	 TRIM: greater than 2 deg	Roll and yaw in direction of applied rudder. Pitch up to about 20 deg, followed by pitch down to nearly -90 deg after 180 deg of roll, thereafter pitch attitude oscillates slightly about -30 deg. Lateral oscillations of ±60 deg with 2 to 3 sec period developing after first rolling turn Total maneuver consists of a roll and two turns with a time span of 18 sec and altitude loss of 2500 ft
2. Classic erect spin (to the right)	Fully developed steady state (preceded by Item 1)	 TRIM: same as Item 1	Pitch attitude: Oscillate between -25 deg and -35 deg with 2 to 3 sec period Roll attitude: With 1/3 left lateral input oscillations of less than ±20 deg with 2 to 3 sec period. Amplitude increase when left lateral control input increased Yaw rate: Steady, 3-1/2 to 4 sec per turn Angle of attack: Cockpit indicator pegged at 30 units (actual angle of attack varies between 75 and 85 deg) Airspeed: Fluctuates from 50 to 150 KIAS, may sometimes reach 185 KIAS Altitude loss: 1100 to 1500 ft per turn Turn needle: Pegged in spin direction
3. Optimum recovery technique	Recovery to level flight (from right spin, Item 2)	 TRIM: 4 deg	Yaw rate: Goes to zero within 1-1/2 turns Pitch attitude: -70 deg Lateral control to neutral after turn stops Altitude loss: 4000 to 5000 ft with airspeed at 220 to 250 KIAS and normal acceleration at 2 to 2-1/2 g Buffet: Occurs during dive recovery, depending on angle of attack/Mach number conditions
4. Effect of longitudinal trim on entry	Entry	 TRIM: -1 deg	Post stall gyrations, no spin
5. Effect of longitudinal trim on recovery	Recovery to level flight (from right spin, Item 2)	 TRIM: 10 deg	All parameters same as Item 3 except that a push force of approximately 10 lb required to attain neutral longitudinal stick position and to maintain dive recovery parameters
6. Effect of neutralizing all controls	Recovery to level flight [from spin to right (Item 2) or left]	 TRIM: 4 deg	Yaw rate: Goes to zero within 3 to 5 turns Other parameters: Same as Item 3

TABLE A-2 - ERECT SPIN MODE TESTS (Continued)

Test item	Maneuver phase	Control input	Expected response
7. Effect of aileron A. Toward spin direction	Initial recovery (from right spin, Item 2)	 TRIM: any	Yaw rate: Goes to zero in 2 to 3 turns
B. Neutral		 TRIM: any	Yaw rate: Goes to zero in 4 to 6 turns
8. Effect of rudder	Initial recovery (from right spin, Item 2)	 TRIM: any	Yaw rate: More than 2 turns required before reaching zero
9. Effect of elevator	Steady state spin (during right spin, Item 2) TRIM: any		Yaw rate: Increase by approximately 20 deg/sec Roll attitude: Oscillation amplitude increases slightly

Note: The left erect spin, including control effects, is symmetric to the right erect spin.

Table A-3 - DIVING SPIRAL MODE TESTS

Test item	Maneuver phase	Control input	Expected response
1. Typical diving spiral (to the right)	Entry and steady-state (controls applied at 20 to 21 units AOA)	 TRIM: any	Initial response Roll and yaw in direction of applied rudder Steady-state response Angle of attack: Varying between 20 and 30 units (up to 70 deg). Pitch attitude: Approximately -50 deg (varying between -30 deg and -90 deg). Roll attitude: Oscillatory (amplitude = ±45 deg, period = 2 sec). Oscillation amplitude increases to maximum of ±70 deg as airspeed increases Yaw rate: Hesitant and never greater than 40 deg/sec Indicated airspeed: Fluctuating and increasing to 220 KIAS
2. Optimum recovery	Recovery (from Item 1)	 TRIM: any	Recovery within 3 to 4 sec (yaw rate and wing rock go to zero) followed by normal dive recovery to level flight.
3. Effect of rudder (applied opposite to turn direction)	Recovery (from Item 1)	 TRIM: any	Recovery within 3 to 4 sec (yaw rate and wing rock go to zero) followed by normal dive recovery to level flight.
4. Effect of elevator	Recovery (from Item 1)	 TRIM: any	No change, diving spiral should continue
5. Effect of aileron	Entry (Controls applied at 20 - 21 units AOA)	 TRIM: any	Initial response: Slow roll in direction of applied aileron Steady-state response: Typical diving spiral in direction of applied aileron after two initial rolls
6. Effect of aileron and rudder combination (Controls applied at 20 to 21 units AOA)	A. Entry and steady state TRIM: any		Enter typical diving spiral to right with occasional transition to erect spin for 1 to 3 turns
	B. Entry and steady state TRIM: any		Enter diving spiral to right with violent roll oscillations (roll rates up to 200 deg/sec)

Note: Left diving spiral, including control effects, is symmetric to right diving spiral.

81-0968

THE NORTHROP F/A-18L MISSION SIMULATOR

RICHARD A. WEEKS
Northrop Corporation
Hawthorne, California

Abstract

This paper presents the current status of the Northrop Corporation F/A-18L Mission Simulator. The F/A-18L Mission Simulator is a real time, man-in-the-loop system simulation of the Northrop F/A-18L aircraft and its avionics/flight control systems. The primary objective of this simulation is to develop, implement and verify the integration of the F/A-18L avionics systems with the human pilot. This objective is being fulfilled by the evaluation of a simulated F/A-18L crew station in a realistic visual environment for specific Air-to-Air and Air-to-Ground mission profiles.

The F/A-18L Avionics System Simulator utilizes a fixed-based visual flight simulator with a high fidelity cockpit environment. Provided in the simulation are detailed simulations of the operational characteristics of the aircraft's Radar, Head-Up Display, integrated stores management, mission computer and navigation systems. The F/A-18L Avionics System Simulator also utilizes detailed models of the airframe and flight control systems.

Introduction

Due to the complexity of current and next generation fighter aircraft avionics sub-systems, the need for detailed modeling and simulation of these avionics sub-systems has developed. From these simulations, the design can be conveniently studied and then refined into a form in which representative hardware and operational flight software can be generated.

However, sub-system simulation is not enough. Because of the intricate and integrated nature of sophisticated fighter aircraft systems, an exhaustive real time man-in-the-loop total aircraft system simulation is required. In this type of simulation, the system concept is of paramount importance. Furthermore, the system is one in which the human pilot or operator plays a crucial role in the design. It is at this level, where designer and user talents combine to yield an operationally effective product.

Overall Simulation Program

In order to effectively study the integration of the F/A-18L Avionics Systems, an extremely detailed high-fidelity crew station had to be constructed. For this effort, actual aircraft instruments/controls were procured, when available, and modified in-house for simulator use. Cockpit geometries were replicated accurately relative to the actual aircraft design as well as the lighting of the cockpit interior and instrument/control panels. The electronic displays and electronics utilized in the cockpit were developed in-house since operational display units did not lend themselves to easy integration with general purpose graphics processors which are required for efficient developmental display concept studies.

While investigating the effectiveness of the pilot in operating the Avionics sub-systems in the cockpit, a fairly realistic threat/visual environment had to be provided. The visual environment was provided by use of the Northrop Corporation Visual Flight Simulator. Visual cues provided in this simulator included an earth/sky image and adversary target image. The simulator hardware configuration is discussed in more detail in the following Simulator Facilities section.

For the threat environment, a multi-airborne target simulation was constructed. Six independently moving target models were utilized which were Radar viewable. Only one of these targets, however, could be acquired visually and was then projected as the visual adversary target image described above. Furthermore, one of these targets could assume a fully interactive attack capability. This interactive target performs logical, tactical decisions based on the current attack engagement geometry and states.

For the avionics system simulation provided in the F-A/18L Mission Simulator, the operational characteristics of the actual avionic sub-systems were implemented. The avionics system is primarily identified by 3 distinct master modes: Navigation, Air-to-Air and Air-to-Ground. These modes are described in detail in the sections which follow. To complement the avionics simulations, detailed models of the F/A-18L flight control system and aerodynamic characteristics, including post-stall aerodynamics, were incorporated in the simulation to include any effects of airframe-coupled sensors on avionics systems performance.

Simulator Facilities

The simulator facilities used to support the F/A-18L Avionics System Simulator consists of a Visual Flight Simulator (VFS) and supporting computer facility. The VFS is a device in which the simulated crew station is supported by a fixed pedestal which is centered within a 24-foot spherical dome (See Figure 1). The interior of the dome acts as a screen for a projected visual target, earth/sky and terrain images. As previously mentioned, the crew station accurately replicated the actual F/A-18L cockpit design. Figure 2 shows the detailing of the simulated crew station.

The visual target is produced by a high intensity projection of an image which is generated from viewing a 3-dimensional target model with a closed circuit television camera (CCTV). The target model is gimbaled and computer controlled so as to continuously provide proper line-of-sight geometries and attitudes to the pilot. Ranging functions are

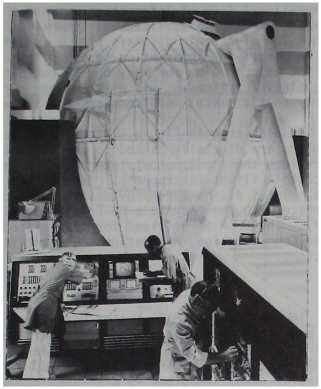


FIGURE 1. VISUAL FLIGHT SIMULATOR

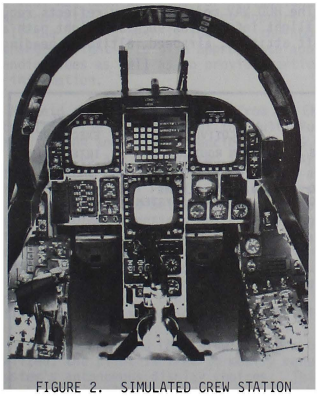


FIGURE 2. SIMULATED CREW STATION

produced by a zoom lense on the CCTV. The target image is blanked beyond visual range and blinks when a theoretical kill is achieved.

The earth/sky image is provided by a single projection system which is located above and behind the pilots head. A single transparency projects a read-brown earth hue whereas another transparency projects a blue shade for sky references. Along the lower portion of the sky transparency is a mountain reference which allows for accurate turn-rate cueing. The earth/sky projection system utilizes a 4-axis gimbal system which permits a continuous attitude reference, free of gimbal lock.

A new terrain projection system consists of 3 projectors mounted on a platform behind the pilots-head and below the earth/sky projector. The terrain image is produced by a Digital Image Generator (DIG) which provides extremely detailed landmass and ground-based cultural features. Figure 3 shows typical terrain scenes provided by the DIG. When all 3 projectors are utilized, a 50 degree vertical by 150 degree horizontal field-of-view terrain image is obtained.

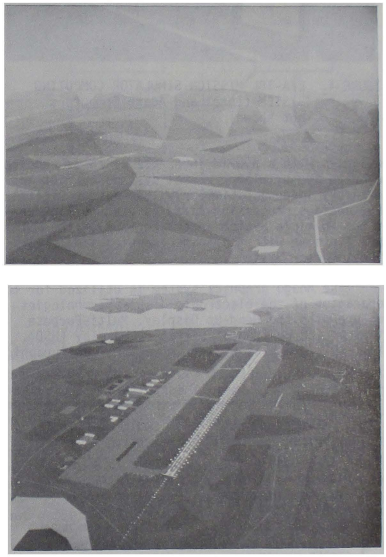


FIGURE 3. DIGITAL IMAGE GENERATOR TERRAIN SCENES

The second major facility utilized in the support of the F/A-18L Mission Simulator is a general purpose computing system. This computing system is configured primarily with Harris Slash 4 minicomputers, Floating Point Systems AP-120B Array Processors, Adage 4135 Vector graphics systems and Intel 8080 microcomputers as shown in Figure 4. The Harris "5-plex" (which consists of five Harris Slash 4 minicomputers) is configured with a Master Digital Central Processing Unit (CPU) which utilizes a Virtual Memory, Master Hybrid CPU and 3 slave CPUs, all of which are tied to a shared common memory. The avionics, aerodynamics, flight control and other related software modules are partitioned within the Harris 5-plex as shown in Figure 5. In addition to the processing functions which are shown in Figure 5, the Master Hybrid CPU com-

After Radar mode initialization, the pilot is free to change modes as desired. Besides the RWS mode, the Radar can be operated in a Track-While-Scan (TWS), Velocity Search (VS), Single-Target-Track (STT), Auto-Acquisition and Specialized Weapon Delivery modes. In the TWS mode, the Radar continuously scans for targets while it keeps track of previous target returns. Once the Radar has assessed the presence of threats, the system will automatically prioritize the threats and will assign a target as a launch and steering (L&S) target. The pilot, however, can override the system selection of the L&S target. The VS mode searches for targets with velocities which are faster than the "ownship".

Once a target is detected, it can be manually "locked on" to place the Radar into the STT mode. In STT the Radar display continuously provides target location in azimuth and elevation, as well as closing velocity, differential altitude and target heading/acceleration information. The STT mode can also be entered via autonomous lock-on by one of the Radar's auto-acquisition modes.

The SMD rather simply shows the selected station for weapon launch as well as weapon quantities available and weapon status. Gun firing rate and gun algorithm utilization can also be manually selected from the A/A SMD.

Air-To-Ground Master Mode

The Air-to-Ground (A/G) Master Mode is entered from the Master Mode panel which brings up A/G attack symbology on the HUD, A/G Radar on the RMPD and A/G Stores Management on the LMPD. Weapon options available in the A/G mode are displayed at the top of the Stores Management Display (SMD). When the A/G Master Mode is initially entered, the left-most displayed weapon option is selected as the priority weapon. The pilot, however, is free to reselect any weapon at his discretion.

When an A/G weapon is selected, either manually or autonomously, a weapon delivery program appears at the center of the SMD. This program is usually established prior to takeoff but can be entered or modified in-flight. Program parameters characteristic to the A/G delivery include: Mode (continuously computed impact point, continuously computed release point, manual, etc.), fuzing (mechanical and electrical), drag options, quantities, multiples, intervals and terrain clearance altitudes. Each weapon option allows for three separate weapon delivery programs. Therefore, the pilot merely has to step through the programs to select this delivery mode rather than continually modify a single program as the weapon delivery scenario changes unexpectedly. Other information portrayed on the SMD are the weapon quantities available and their location as well as the selected weapon station for release.

As mentioned above, A/G Radar is initialized on the RMPD upon initial A/G Master Mode activation. If an A/G target has been previously designated, the Radar enters an A/G Ranging mode. If no A/G target is formally designated, then the Ground Mapping mode of the Radar is entered.

Summary

This paper has presented a very brief overview of the F/A-18L Mission Simulator program. The detailing of the results of the evaluations conducted in the F/A-18L Mission Simulator are well beyond the scope of this paper. In summary, an effective real time man-in-the-loop simulation program has been successfully utilized to investigate the integration of the sophisticated avionics systems in the F/A-18L aircraft. Having considered the human pilot with these sophisticated electronic systems has allowed early evaluations of potential problem areas which have resulted in numerous significant design changes in the avionics systems. These changes have been primarily in the area of the electronic display symbologies and associated switchologies. Furthermore, these changes have made the pilot's job a little easier which ultimately leads to safety and survival.

Aknowledgement

The author wishes to acknowledge the contributions and efforts of the many people in the Northrop Aerosciences Laboratory who have been involved in the development and operation of the F/A-18L Mission Simulator program.

D. Hernandez*
North American Space Operations
Rockwell International
Downey, California

Abstract

This paper describes the Space Shuttle hardware evaluator man-in-the-loop system, a real-time simulator designed and built to support the Space Shuttle orbiter vehicle development, checkout, and verification, at the Space Transportation System Development and Production Division of Rockwell International. The simulator is a high-fidelity facility with prototype on-board computers, controls, and displays. An overview is presented of the manner in which this prototype equipment is flown in a simulated environment that is created with general- and special-purpose computers which simulate the orbiter vehicle, the natural environment, and the cockpit visual displays. This paper also describes the simulator system configuration, hybrid computers, and mathematical models, as well as hardware and software implementation features.

Introduction

The Space Shuttle hardware evaluator man-in-the-loop (HE-MIL) system is one of the two real-time, man-in-the-loop simulation facilities located in the Flight Systems Laboratories of the Space Transportation System Development and Production Division of Rockwell International. The development of this facility was initiated in 1974, and the first use of some of its elements was made in the Space Shuttle design and development. In 1977, during the approach and landing test phase of the Space Shuttle, this simulation facility was used to verify the orbiter vehicle's approach and landing after separation from the Boeing 747 aircraft. Presently, it is used to verify the orbit insertion, on-orbit, deorbit, and descent phases of the Shuttle mission.

The HE-MIL system is depicted in Fig. 1 and consists of a fixed-base cockpit, a general-purpose visual display system, a system master control console, a data collection and processing system, general-purpose digital and analog computers, forward and aft interfaces, a flight control hydraulic laboratory, a Shuttle data processing subsystem, and a communication, tracking, and instrumentation station. The system has a full avionic capability, supported by a prototype data processing subsystem and prototype cockpit displays and controls. Using general-purpose analog and digital computers and special-purpose hardware, it simulates the orbiter vehicle, the natural atmospheric and orbital environment, and the crew out-the-window landing visual scenes. When more fidelity is required in the orbiter vehicle simulation, this system is also supported by a prototype communication and tracking subsystem, instrumentation subsystems, and a hydraulic aerosurface actuation subsystem.

Hardware Evaluator Man-in-the-Loop System

Major Capabilities

The HE-MIL system is a large and complex simulator with a significant number of capabilities. The major capabilities of the mission coverage (those mission segments simulated by the HE-MIL system), the simulated natural environment, the Shuttle systems simulation, and the simulator data, failure, and visual systems, are listed below.

Mission Coverage

- Orbit insertion, on-orbit, deorbit, and entry through roll-out
- Abort once around (after one orbit)
- Glide return to launch site

Natural Environment

- Kennedy Space Center (KSC), Edwards Air Force Base (EAFB), and 1962 standard atmospheres
- KSC and EAFB winds
- Spectral and discrete turbulence
- Earth 1960 Fisher ellipse model
- Runway model

Vehicle Dynamics

- Six-degree-of-freedom orbiter
- Three-degree-of-freedom external tank
- Entry and on-orbit aerodynamics
- Aerodynamic data uncertainties
- Variable mass and center of gravity

Data Processing System (Prototype)

- Five general-purpose computers (GPC's)
- Primary and backup flight software
- Master timing unit (MTU)
- Multiplexers/demultiplexers (MDM's)
- Mass memory unit (MMU)
- Display electronics units (DEU's)
- Display driver units (DDU's)
- Multifunction cathode ray tube (CRT) display system

Guidance, Navigation, and Control (GN&C)

- GN&C controls and displays (prototype)
- Three inertial measuring units (IMU's)
- Four rate gyro assemblies (RGA's)
- Four accelerometer assemblies (AA's)
- Four air data systems (ADS's)

*Member, Technical Staff; AIAA Member

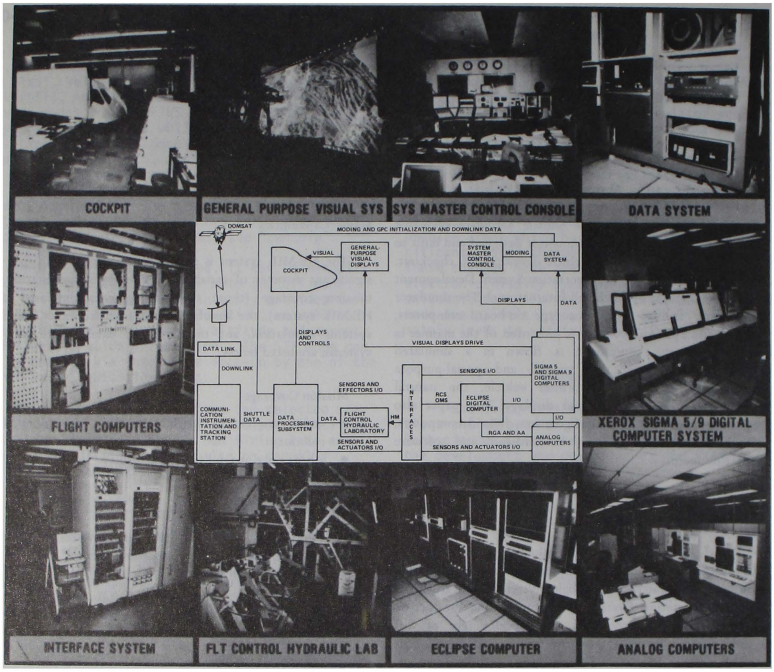


Fig 1 Hardware Evaluator Man-in-the-Loop System

Communication, Tracking, and Instrumentation

- Prototype communication and tracking subsystem
- S-band system
- K-band system
- Pulse code modulation (PCM) master unit
- Payload data interleaver
- Payload signal processor
- Flight recorders
- Network signal processor
- Uplink and downlink telemetry
- Three microwave scanning beam landing systems (MSBLS's)
- Two radar altimeters (RA's)
- Three tactical air navigation systems (TACAN's)

Propulsion

- Reaction control system (RCS)
- Orbital maneuvering system (OMS)

Mechanical Systems

- Hydraulic actuation systems (prototype or simulated)
- Landing gears

Cockpit

- Forward station
- Prototype displays and controls
- Audio effects
- Video tape

Data Collection and Reduction System

- Collect all downlink PCM data
- Collect 1,024 words of simulation data at 25 Hz
- Strip chart and CRT real-time display
- Quick-look data reduction
- Archive data
- Pass/fail real-time system
- Pass/fail offline data reduction

Failure System

- Automatic or manually injected failures
- Multiple failures per run

Visual System

- Closed-circuit, color television system
- Forward commander window view

The HE-MIL system can simulate a continuous mission from external tank separation through roll-out, but does not simulate the ascent mission phase prior to external tank separation.

The simulated natural environment provides the natural atmospheres, winds, and turbulence experienced by the vehicle during the simulated mission phases. The earth's shape is math-modeled with a 1960 Fisher ellipse, and the landing field with a concrete or lakebed runway.

The orbiter vehicle dynamics are math-modeled with six-degree-of-freedom equations of motion, variable mass, and variable center of gravity. Its motion can be simulated with Newtonian translational equations of motion in the body or inertial frame and with quaternion or Euler rotational equations of motion. The on-orbit and descent aerodynamics are modeled, but the ascent aerodynamics are not. No vehicle-bending dynamics are modeled.

The HE-MIL system has a prototype data processing subsystem with all the elements contained in the HE-MIL Major Capabilities List. The five on-board general-purpose computers are loaded with actual flight software. The subsystem interfaces with the simulation computers through the MDM's and the PCM units.

Simulated redundant inertial measuring units, RGA's, AA's, and air data systems sensors are also contained in the HE-MIL system. Except for the inertial measuring units, all simulated sensors have a single math model with multiple fanouts, that have the capability to introduce failures, noise, and errors in the fanout loops. Prototype rate gyros and accelerometers can be used in place of the math models.

When needed, a prototype communication and tracking subsystem with all the elements contained in the HE-MIL Major Capabilities List can be connected to the system. A Shuttle avionics test set (SATS) is used to provide the uplink capability. The microwave landing systems, the radar altimeters, and the tactical air navigation sensors are simulated with a single math model and multiple fanouts, with the capability to introduce failures, noise, and errors in the fanout loops.

The RCS and the OMS are simulated. Since ascent is not simulated, the main engines and the solid rocket boosters are not modeled.

Math model aerosurface actuators or prototype actuators can be used. In the closed-loop simulation, the Flight Control Hydraulics Laboratory provides hydraulic systems, aerosurface actuators, simulated aerosurfaces, and aerosurface load actuators.

The cockpit has a fixed base and contains the forward station with prototype displays and controls. It has audio effects during thrusting and roll-out. Failures are implemented only on the left side of the cockpit.

The data system can collect all the downlink PCM data, and 1,024 words, each made up of 16 bits, from the simulated vehicle and natural environment 25 times a second. It then merges and writes the data on digital tapes for offline data reduction. In addition, it has a NOVA 840 computer that performs online pass/fail calculations, as well as stripping of PCM and environment parameters at 1 Hz for offline processing on IBM or Data General computers. There are 88 channels of continuous analog strip recording, of which up to 32 channels can be PCM downlist parameters.

The failure system is automatic and can set or reset over 4,000 different failures, of which up to 225 can be set in a given run. They can be programmed to occur within one minor cycle (20 milliseconds) of each other, with no restriction on the frequency of occurrence.

The visual system is a color television, six-degree-of-freedom system with a full view out the left forward window. No displays are provided for the side or quarter windows. During approach and landing, this system provides a virtual image of the runway.

Basic Configuration Description

The HE-MIL system configuration is depicted in Fig. 2, which is a simplified block diagram that shows only the most important elements and interfaces in the system.

The digital and analog computers—the Xerox Sigma 5, Xerox Sigma 9, Data General Eclipse, Digital Equipment Company PDP 11, Texas Instruments (TI) 990 microprocessor, and six Electronic Associates, Incorporated, analog computers—simulate the natural environment, plus a Shuttle six-degree-of-freedom airframe, avionic sensors, effectors, actuators with loads, cockpit display functions, and visual display drive equations. The natural environment and vehicle math models are solved in real time and with enough fidelity to interact with the on-board flight computers through the forward and aft MDM's. The interaction occurs in such a way that it provides a test bed for the performance verification of the GN&C system in either the manual or automatic mode. The cockpit display functions are solved in the Sigma computers for display in the cockpit and the system master control console (SMCC). The visual display drive equations are solved in the Sigma and PDP 11 computers in order to drive the general-purpose visual system (GPVS) transport and optical probe. The OMS and aerosurface actuators are math-modeled in the analog computers. The OMS and RCS thrust, forces, moments, and fuel accounting are modeled in the Eclipse computer with the help of four data control units (DCU's) and a TI 990 microprocessor. In addition, the hybrid computers calculate all the data required for run evaluation and validation, then transmit that digital and analog information to the data system (this transmission is depicted with broken lines in Fig. 2) through the Sigma/Nova interface and the analog-to-digital converters (ADC's), respectively.

The HE-MIL system can run with prototype or simulated systems in various areas. It can, for example, run with prototype or math-modeled aerosurface amplifiers and actuators. The prototype aerosurface actuators are located in the Flight Control Hydraulic Laboratory (FCHL). The Sigma computers supply the aerodynamic hinge moments to the aerosurface actuation models or to the FCHL load actuators, in either configuration. In addition, the hybrid computers simulate the rate gyro and accelerometer sensors, or supply drive signals when the vehicle equivalent hardware sensors are used.

The HE-MIL system uses several special-purpose simulators with the hybrid GPC's and visual system. A valve simulator subsystem that simulates the fuel and oxidizer valves first accepts commands from either the on-board GPC's or the cockpit and

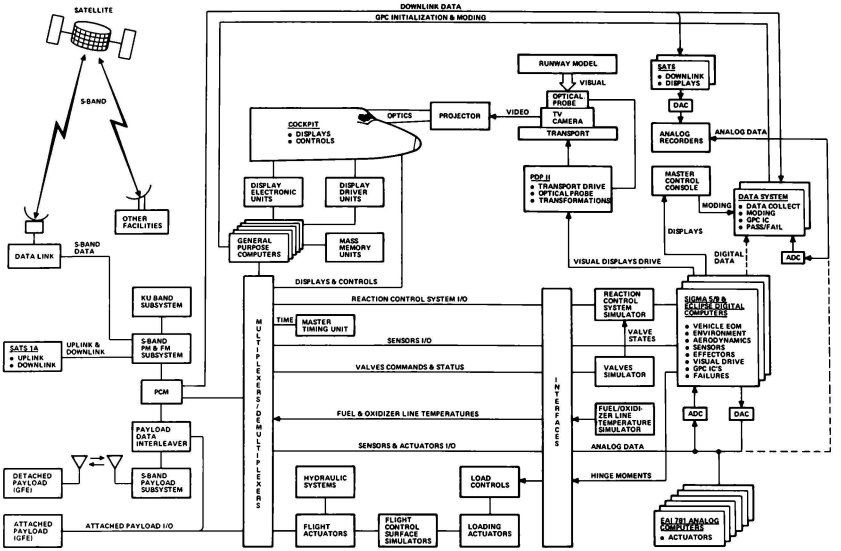


Fig 2 HE-MIL System Configuration

then provides valve status to the GPC's and simulation digital computers. The simulation computers use the valve status information to determine which RCS jets or OMS engines can be fired, and to determine the fuel and oxidizer consumption by tank. A fuel and oxidizer line temperature simulator then provides temperature information to the GPC's through the forward and aft MDM's. This is an open-loop simulator, and variations and failures in these temperatures can be introduced manually into it. An RCS simulator uses a TI 990 microprocessor to accept the jet on/off commands, simulate the jet thrust forces and chamber pressures, and provide the chamber pressure discretes and driver on/off status to the GPC's. It also simulates the timing, thrust shaping, and jet failure logic, and provides the timing synchronization between the GPC jet commands and the computer math modeling of the vehicle and sensors. An MTU emulator that can be used in place of the prototype unit and also started, stopped, and initialized directly by the data systems computer is also a part of the HE-MIL system. When the prototype MTU is used, an MTU start/stop interface box that is controlled by the data system must be used, too. Other special-purpose simulators like the failure system—which introduces sensors, actuators, controllers and switch failures—and the cockpit and related visual system are also used by the HE-MIL system.

The hardware evaluator test station contains a data processing system, the forward and aft interface units, the GPC interface assemblies (GIA's), and the mode controller. The data processing system components listed in the HE-MIL Major Capabilities List are prototype and their function and operation are similar to those of the vehicle equivalent components. The other elements in this station are not prototype and are required to connect the data processing system to the simulated Space Shuttle subsystems. The forward and aft MDM's communicate with the hybrid computers through the forward and aft interface units. These units transform data back and forth between the hybrid computers' digital and analog domains and the MDM's serial Manchester, digital discrete, or analog format. The GIA's are the interfaces that provide the capability to load, initialize, and mode the GPC's from the data system. The mode controller provides the system configuration and moding interface to the different systems or computers.

The data system collects all the pertinent data required for system evaluation and verification, modes the stations or computers in the HE-MIL system, and loads and initializes the GPC's by using a Data General Nova 2 and a Nova 840 computer. Up to 1,024 16-bit words of vehicle and environment information are collected by the Nova 2 computer through the Sigma/Nova interface, as well as all of the downlist PCM data through the PCM decommutator interface. All of this information is then written on magnetic tapes for data reduction and processing in the data reduction systems at a later time. In addition, the Nova 2 has the moding control program that is controlled from the SMCC. This program controls the overall system moding through the Nova interface assembly (NIA) and the mode control console, using the GIA's to mode the GPC's and the independent moding terminals to control the other computers or stations in the system configuration. During the initialization mode, this same Nova 2 can either load the GPC's with full memory loads or initialize any location in the GPC memories with data provided by the Sigma computer equations or by the test operators. The Nova 840 in the data system is used to calculate the pass/fail equations or criteria

and it also strips downlist and environment parameters at 1 Hz for offline processing on IBM or Data General computers. The Nova 840 also reads analog variables using analog-to-digital converters and stores them on tape for offline processing.

A complete description of each of the HE-MIL system elements is found in Refs. 1 and 2. The HE-MIL system is part of the Flight Systems Laboratories, which are described in Refs. 2 and 3.

Hybrid Computing System

The HE-MIL hybrid computational system consists of three digital computers—a Xerox Sigma 9, a Xerox Sigma 5, and a Data General Eclipse—six EAI 781 analog computers, and the associated input-output devices. The Xerox Sigma 5 and Sigma 9—sometimes referred to as the Sigma 5/9 system—are high-speed, general-purpose digital computers with the necessary input-output devices for real-time interaction with the external analog and discrete world.

A basic Sigma system has a central processing unit (CPU), a main memory subsystem, and an independent input-output subsystem, all of which perform asynchronously with respect to each other. Arithmetic units, real-time clocks, external interfaces, interrupt control chassis, and memory write protection are all included in the CPU's. Floating-point instructions are available in both single-precision (32-bit) and double-precision (64-bit) formats.

The main memory subsystem consists of four modular units of 32,768 words each; a unit consists of two banks of 16,384 words each. A multiport system in each memory unit allows independent access to the memory by five processors—two CPU's, one input/output processor, and two direct-memory-access processors. Each memory word has 32 bits.

Communications with the external analog and discrete world are conducted by way of two direct-memory-access systems and a direct input-output system. Two channels from the first direct-memory-access system are used for the ADC's and digital-to-analog converters (DAC's), which interface with the cockpit, and a third channel is used for digital-to-digital (D/D) communication with the HE forward interface and the visual system PDP 11 computer.

Two channels from the second direct-memory-access system are used for the 64 ADC's and 64 DAC's that interface with the analog computers. The direct input-output (I/O) system interfaces with three, 64-bit digital I/O terminals and a subsystem with 72 external interrupts.

The Eclipse system centers on a Data General Eclipse S-200 CPU (16-bit word minicomputer) with memory mapping and protection (128K-words, four-way interleaved), a 64-bit floating-point processor, a real-time clock, and priority interrupts. To augment the Eclipse computer throughput, four Data General DCU's with multiply-divide hardware have been incorporated into the system. The DCU's are normally dedicated communications controllers but in this system they are dedicated to table interpolation, which results in a substantial reduction of Eclipse CPU computational time usage. This system has 16 ADC's and 16 DAC's. It also has digital-to-digital data links to a TI 990 microprocessor and the Sigma 9/5 system.

Presently, the HE-MIL system uses six sections of EAI 781 analog computers that are supported by an EAI Pacer 100 digital computer. A typical EAI 781 analog section has 42 integrators, 66 summers, 24 multipliers, 4 resolvers, 6 digital function generators, 24 comparators, 24 digital-to-analog (D/A) switches, 24 function relays, 144 servo attenuators, 24 limiters, and 72 digital controlled attenuators (DCA's). In addition, each analog section has a general-purpose logic system with 56 gates, 6 registers, 6 counters, 2 timers, 36 flip-flops, 4 logic pushbuttons, and 6 differentiators.

Math Models

Hybrid computer math models are used to simulate the natural environment, Shuttle airframe, sensors, effectors, actuators with loads, cockpit display functions, and visual display drive equations. A simple block diagram of the math models is shown in Fig. 3. These models are solved in real time at speeds high enough to prevent a significant degradation in the performance of the pilots, the vehicle, the sensors, or the solutions of the on-board GPC's. In addition, solutions are performed in the analog or digital domain, as dictated by interface or frequency response requirements. The real-time models produce enough information for real-time data collection and non-real-time data reduction to evaluate the performance of the vehicle systems. The math models can be configured to support static or dynamic tests, open-loop or closed-loop solutions for point stability, or orbiter six-degree-of-freedom analyses. The environment math models can be initialized in the landing field, inertial, or heading frame of reference, and they produce enough parameters to fully define an initialization state in all the frames not directly initialized.

The orbiter equation-of-motion math model is a six-degree-of-freedom model. The simulated spacecraft is flown over an oblate, rotating earth with a gravity field, wind and gust effects, and realistic atmospheric properties. The landing field or runway can be positioned anywhere on the earth, and crown and wet and dry effects can all be simulated. Next to a scaled runway model, a closed-circuit camera with a probe and transport system is driven by a math model to provide an out-the-window view compatible with the relative positions and orientations of the runway and spacecraft. The aerodynamic model and data for the orbiter are taken from the Aerodynamic Design Data Book (Ref. 4). Orbiter rotational and translational motion is simulated using the mass properties of the particular vehicle. The elevon, rudder/speed brake, and body flap actuators, with their associated aerosurface amplifiers, are math-modeled in the analog domain.

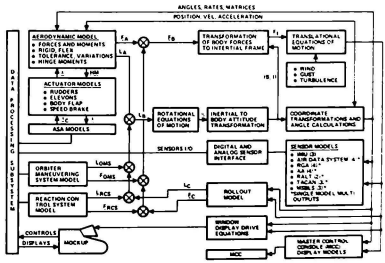


Fig. 3 HE-MIL Math Model Overview

The hinge moments produced by the aerosurface interaction with the environment are math-modeled for the software or hardware actuators.

The avionics sensors are math-modeled and driven to satisfy the MDM interface requirements. The IMU, RGA, ADC, and AA are math-modeled with a 20-ms frame cycle time (the fast cycle) and the RALT, TACAN, and MSBLS with a 40-ms cycle. Due to computer-equipment and frame-time execution limitation, only the IMU has a full redundant set simulated, while the remaining sensors have some elements simulated with output fanouts that satisfy GPC self-test commands and MDM interface requirements. At present there is the capability to drive (torque) prototype RGA's and AA's. The aerosurface amplifiers are simulated when the simulated aerosurface actuators are used. The roll-out model has antiskid logic, differential braking, nose-wheel steering, a gear model, and the contact forces and moment equations necessary to interface with the runway model.

The RCS model provides forces, moments, and propellant accounting, as well as feedback to the GPC's. Also provided are plumbing, thrust shaping, jet impingement on aerosurfaces, and aerodynamic interaction. Since some of the plumbing is shared with the OMS, a common model is used.

The OMS is a hybrid model with pitch and yaw actuators mechanized on the analog computers, while thrust shaping, forces, moments, and propellant accounting are mechanized on the Eclipse computer.

In addition to providing environment/vehicle models, the hybrid computer program is required to provide other closed-loop support functions. Firstly, it must solve the equations necessary to drive the visual display system so that it provides the commander's window display. Secondly, it must provide the signals necessary to drive the simulated display, both in the cockpit and on the master control console. Thirdly, it must have a moding program capable of being controlled from the closed-loop moding system. And, finally, it must initialize locations in the GPC's with data derived from the hybrid computer models or from initialization data supplied by the sponsors.

An overview of the math models, their computational flow, and their relative timing is depicted in Fig. 4. This figure emphasizes the main math models, their relationship to each other, the analog, digital, and hardware domain boundaries, and the main computational delays. The boundaries between the digital and analog models are depicted with a sampler for an ADC, and with a sampler and a zero-order-hold (ZOH) for a DAC. The interface between the simulated environment and the on-board data processing subsystem are the MDM's, which are shown at both ends of the figure. The computational delays are indicated with the z-transformation symbol Z^{-1}. Predictors are used where required to compensate for delays caused by parallel processing, lack of computational speed, feedback data transmittal, and D/A conversions. Fig. 4 is primarily descriptive and therefore not a precise rendering of the HE-MIL math model computational flow. There are nine digital processors involved in the solution of these models, and the computational split and sequence among them is very complex. Ref. 1 shows a computational sequence diagram that illustrates the parallel operation, synchronization, and data transfer among the digital processors. A full description of the math models with signal flow diagrams and implementation algorithms is found in Refs. 1 and 5.

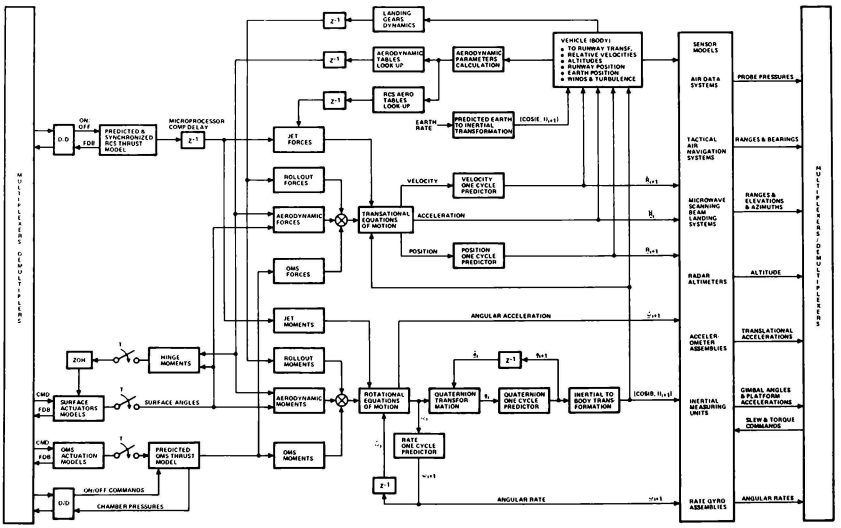


Fig 4 Math Models Computational Flow Overview

Math Model Problem Split

The HE-MIL system was designed to be a real-time engineering developmental tool and subsequently became a vehicle and system verification facility. It was therefore decided that a significant amount of prototype hardware would be used along with a hybrid computer facility to meet the high-fidelity simulation requirements of the HE-MIL system. The prototype elements used are: the data processing subsystem; the communication, tracking, and instrumentation (CT&I) station; and the FCHL actuator systems. These elements were built into separate stations for independent and simultaneous use as stand-alone subsystems, or for combined use in the HE-MIL closed-loop system. The rest of the required Space Shuttle systems were simulated with a general-purpose digital multiprocessor system, six EAI 781 analog computers, and some special-purpose simulators.

The math models and problem allocation among the different computers are shown in Fig. 5. This figure depicts only the hybrid computers and interfaces, and therefore does not show the GPC's, the CT&I, and the FCHL elements. Many interrelated factors were considered in allocating the different models to the different types of computers (digital computers, minicomputers, microcomputers, and analog computers) or special-purpose simulators. Most important among them were implementation cost and time, ease of implementation and checkout, input and output requirements, kinds of model equations, modeling precision, frequency and timing requirements, and the capabilities and limitations of the different computers and their interfaces. The digital computer coding had to be done in assembly language because of the magnitude and complexity of the models, the 20-ms basic cycle modeling requirements, and the relatively limited computational power available.

General-Purpose Visual Display System

A simulated out-the-window scene that accurately simulates the real-world scene is provided for the pilot by the general-purpose visual display system. Although it produces a full-field-of-view color television display in the forward window, it does not provide one for the side or quarter windows.

The visual system consists of a field-sequential color television camera and projector, an optical scanning probe, a three-degree-of-freedom camera transport, a virtual image display system, a high-altitude and a landing model, and a PDP 11 digital computer and interface system. Operating with its television standard of 625 lines per frame, 180 fields per second, the system's camera and projector provide a static out-the-window resolution of approximately six arc minutes. The 120-degree field-of-view optical probe used by the system to provide three rotational degrees of freedom, along with the camera transport used to provide three translational degrees of freedom, functions under computer control to maneuver the television camera and optical probe above the terrain models to simulate vehicle translations and rotations. The virtual image display system consists of a collimating window lens, a rear projection screen, and a light shroud. This system conveys a realistic impression to the pilot commander because the image appears to be a considerable distance from the cockpit.

Two models are used in the visual display system: a landing model and a high-altitude model. Both of them depict the runway and surrounding terrain at KSC. The landing model is 15 feet by 52.5 feet, with a scale factor of 1,500 to 1, and the high-altitude model is 5 feet by 10.5 feet, with a scale factor of 1 nautical mile per inch. The landing model is basically used at altitudes below

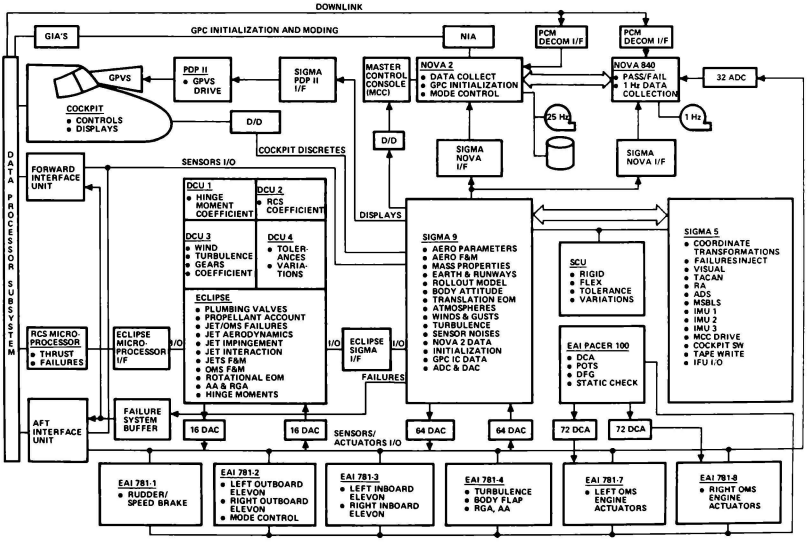


Fig 5 HE-MIL Models and Problem Allocation

10,000 feet, while the high-altitude model is used above 10,000 feet.

The visual display system uses a dedicated PDP 11 digital computer to transform vehicle attitude and translation data into command signals for the visual system hardware. The PDP 11 accepts commands from the Sigma 5/9 computers, provides command signals to the optical probe and transport servos, and receives position feedbacks from the transport.

Data Processing Subsystem

The data processing subsystem provides data processing capabilities for GN&C, communications and tracking, displays and controls, system performance monitoring, payload management, payload handling, subsystem sequencing, and selected ground functions. This subsystem accepts input commands and data from the crew, on-board sensors, and external sources; it performs computations and processing, and generates output commands and data as necessary to accomplish the on-board data processing requirements.

The digital data processing subsystem consists of a main computer complex of five GPC's and assorted interface units interconnected by seven groups of digital data buses. The GPC processing elements are the CPU, which provides the central computational capability, and the input-output processor (IOP), which provides centralized control and interface linkage between the associated GPC memory and interfacing subsystems. The main interface units between the data processing subsystem and the simulated Space Shuttle subsystems are the MDM's.

The MDM's act as data acquisition, distribution, and signal conditioning units. They accept serial digital information from their controlling units and convert or format this information into analog, discrete, or serial digital form for transfer to the simulated Space Shuttle subsystems. The MDM's also receive analog, discrete, and serial digital information from the simulated Space Shuttle subsystems and convert and format these data into serial digital words for transfer to their controlling units.

Flight Control Hydraulics Laboratory

FCHL provides the test bed for the vehicle hydraulic components and subassemblies. The facility provides hydraulic systems, aerosurface actuators, simulated aerosurfaces (inertia, weight), compliances, and aerosurface load actuators, and may be operated independently, open loop, or in the HE-MIL closed-loop environment. The FCHL layout, including plumbing, is arranged to duplicate the relative locations of the hydraulic interfaces within the limitations of the building.

The elevon servoactuators, body flap actuator subsystem, and the rudder/speed brake subsystem are orbiter hardware. Except for the drivers, which are variable-speed electric motors rather than auxiliary power units, the power system components are also principally orbiter hardware.

Each flight control surface inertia is simulated with an equivalent mass, and the aerodynamic loading is provided by a separate hydraulic servo-controlled loading system. The loading is a function of the hinge moments, which are calculated in the hybrid computers and supplied to the load control system shown in Fig. 2.

Communication, Tracking, and Instrumentation

The CT&I laboratory facility is used for development, checkout, and verification of the orbiter- and payload-related communication and tracking hardware. The main components of this facility are listed in the Major Capabilities List and organized around various test substations that can be used to stimulate and/or monitor each line-replaceable unit when it is independently tested or when it forms part of a system. This station is capable of handling radio-frequency navigational aids, ultra-high-frequency, audio, S-band, and Ku-band communications, and Ku-band radar.

Of all the substations within the CT&I test station, one of the most important is the Ku-band communication and radar substation. It duplicates the on-board system's capacity to transfer two-way data between the ground and the orbiter, and can play the role of either the ground station or the orbiter during testing in order to check out the system at both ends. Another substation has the S-band communication subsystem, which provides tracking and two-way communication to the ground through a frequency modulation link and through phase-modulated links.

The CT&I facility also houses a high-capacity data link that connects KSC in Florida to the Flight Systems Laboratories in Downey, California. A leased satellite and a telephone land-line carrier complete the link, which allows both sites to record pulse-code-modulated and wideband digital data during tests.

The CT&I station interfaces with the HE-MIL system and other Flight Systems Laboratories facilities to provide a flexible checkout and verification tool.

Timing and Moding

In order to control and synchronize all the different computers and time-tag the collected or transmitted data, a moding and timing system was designed and built. A simplified overview of the HE-MIL timing and moding is shown in Fig. 6. This block diagram shows the various computers, the moding and timing systems, and their related interfaces within the system. Some of the main data communication channels not related to moding or timing are shown with wide lines for a better understanding of the system. The source of the timing and moding is depicted with a heart symbol next to the rubidium clock and the mode control program, respectively.

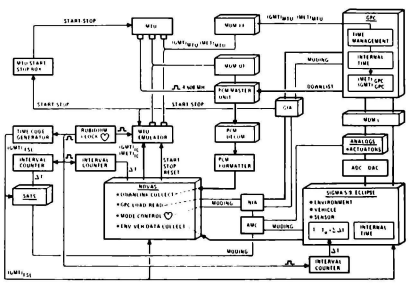


Fig 6 HE-MIL Timing and Moding

The source of the system timing is this high-accuracy rubidium clock, which provides the synchronization pulses to the time code generator, the computer internal counters, and the MTU emulator that is sometimes used in place of the prototype MTU. The time code generator in turn generates the Flight Systems Laboratories Greenwich Mean Time (GMT) serial time code used by the SATS, Nova, Eclipse, and Sigma computers for time tagging or display. The internal counters use a 1-ms or a 1-microsecond frequency pulse to count the computational time interval (ΔT) required by the smallest computational cycle in the digital computers. The counter supplies an interrupting pulse to the host computer every ΔT seconds, after the computer has supplied the base count and the start/stop signal. The MTU emulator supplies GMT and mission event time (MET) serial time code to the MDM's, which then provide the same to the GPC's and PCM master unit. Since the MTU emulator does not provide synchronization frequencies at the present time, they are obtained from the prototype MTU. The MTU emulator is initialized with GMT and MET, and moded directly by the Nova 2. The prototype MTU is initialized by the GPC, and moded by an MTU start/stop through the GIA.

The mode control executive is resident in the Nova 2 and controls the moding, timing, and moding transition of most of the analog and digital computers. The moding signals are transmitted out of the Nova 2 through an NIA, which in turn sends them to the GIA's and the ADL moding control (AMC) unit. The AMC controls the configuration of the system (e.g., which computer or station is in the HE-MIL system) and provides the mode control signals using the moding terminals at each configured computer or station. The mode control system has been developed with both manual and computer-augmented capabilities. The manual mode control is used primarily for checkout of the system, and the computer-augmented mode control is used for closed-loop operations. The system configuration consists of up to 12 stations that can be configured in one or two closed-loop systems.

The station states are:

1. Off-Line – Indicates the station and associated interfaces are electrically and/or mechanically disconnected from the other stations in the configured system.

2. On-Line – Indicates the station is connected to the stations in the configured system.

3. Local – Indicates that the station is under local mode control.

4. System – Indicates that the station is under system mode control.

The system modes of operation are:

1. RIC (ready for initial conditions) – A system status indicating that all the configured stations are ready for problem initialization.

2. IC (initial conditions) – The process of setting up the problem initial conditions. It may be started at any time after all configured stations have issued a station RIC status. Separate IC commands are issued to each station for initialization sequencing purposes.

3. ROP (ready for operate) – A system status indicating that all the configured stations are ready for the operate mode. It is obtained after the individual stations have completed their initialization procedures and issued a station ROP status.

4. Operate – The system problem solution following the start-problem-execution (operate) command. After simultaneous operate commands are issued to all stations, a stop command is issued to all configured stations if operate confirmation feedbacks are not received.

5. Halt – A system status indicating that all the configured stations' executions have been halted after simultaneous halt commands.

6. Stop – The normal stopping sequence after a system problem execution stop command.

7. Emergency – A system status indicating an emergency stop necessitated by an abnormal condition that could cause injury to personnel or damage to equipment.

A transition state diagram of the system moding within the station state domains is shown in Fig. 7. This diagram shows the legal mode transitions, and the required event or conditions for the state transition.

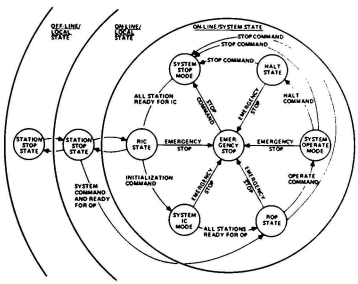


Fig 7 Moding Transition State Diagram

Conclusion

The use of the HE-MIL facility has been greatly enhanced because of its having been built progressively to meet the design, development, and verification stages of the Shuttle orbiter avionics testing requirements. In addition, it was organized into independent stations, each of which could be used as a stand-alone subsystem or as an element of a much larger system. This organization has provided flexibility, maximized resource utilization, and satisfied simultaneous simulation and testing requirements.

The system has made use of prototype vehicle hardware, special-purpose simulators, and a powerful hybrid computation system to function as a high-fidelity verification facility. Some of the prototype hardware can be replaced with simulated elements when the hardware is needed for testing outside the closed-loop environment.

The HE-MIL simulation facility exemplifies a major growth in simulator use, fidelity, and capacity over previous simulators, and it has played a vital role in the design, development, and verification of the Shuttle orbiter vehicle.

References

1. Hernandez, D., et al., *Flight Systems Laboratory Configuration Definition – Math Model Definition*, Rockwell International, SOD 79-0102B (February 1981).

2. Tindal, J.A., et al., *Flight Systems Laboratory Configuration Definition – System Definition*, Rockwell International, SOD 80-0082 (August 1980).

3. Robertson, J. M., and P. D. Liles, *Shuttle Avionics Laboratory*, Rockwell International, SD 77-SR-0028 (September 1977) (AIAA 77-1532).

4. *Aerodynamic Design Data Book, Volume 1*, Orbiter Vehicle 102, Rockwell International Report SD 72-SH-0060-IL (October 1978).

5. Hernandez, D., *Real-Time Hybrid Simulation of the Rotational Motion of a Rigid Spacecraft*, MSEE Thesis, UCLB (1970).

ON-LINE WIND SHEAR GENERATION FOR FLIGHT SIMULATOR APPLICATIONS

81-0970

A. B. Markov*, L. D. Reid† and R. B. MacKenzie††
Institute for Aerospace Studies
University of Toronto
Toronto, Canada

Abstract

A technique for generating on-line wind inputs for flight simulator applications has been developed. This approach yields wind inputs that are produced by a wind controller acting on the aircraft state and input vectors. The form of the wind controller is established off-line through an optimization process that selects its parameters so as to produce winds which represent some type of worst-case situation. Three such wind controllers were tested in a three degree-of-freedom fixed-base simulation of a light STOL transport. Data were collected for steep ILS approaches flown by two pilots, and an evaluation was made of the severity and realism of the generated winds. Comparisons were made with results obtained in the presence of two reference wind profiles. These tests indicate that the wind controller technique has several advantages over existing simulator wind modelling methods and that it should be assessed further in a more sophisticated flight simulator.

Nomenclature

d	glidepath deviation normal to glidepath plane
h	altitude AGL (m)
q	pitch rate (rad/s)
T	natural mode period (s)
$T_{1/2(2)}$	time to half (double) amplitude (s)
u	airspeed component along x stability axis
V_e	reference equilibrium airspeed
W_1	tailwind velocity (ms^{-1})
W_3	downdraft velocity (ms^{-1})
$W1_e$	reference equilibrium value of W_1 (ms^{-1})
$W1_{wc}$	contribution of wind controller to W_1
$W3_{wc}$	contribution of wind controller to W_3
w	airspeed component along z stability axis
x_I	horizontal position
γ_G	glideslope angle (deg)
ζ	damping ratio
θ	Euler pitch angle (rad)
$\mu_{S_{WS}}$	mean of S_{WS}
$\sigma_{S_{WS}}$	standard deviation of S_{WS}
ω	damped natural frequency
ω_n	undamped natural frequency

Notation Conventions

$\underline{X}$	matrix
$\underline{X}^{-1}$	inverse of $\underline{X}$
$\underline{X}^T$	transpose of $\underline{X}$
$\hat{x}$	mean of RMS values of x
$\underline{x}$	vector or column matrix
$\Delta(\ )$	perturbation quantity about a reference equilibrium value

*Research Assistant, Member AIAA
†Associate Professor, Associate Fellow AIAA
††Research Assistant, Student Member AIAA

I. Introduction

A number of landing approach and take-off accidents in which variable winds have been found to be major contributing factors (e.g. the well-documented JFK accident in 1975) have focused attention on wind modelling for aircraft hazard definition.[1] From the perspective of flight simulator wind modelling, such hazardous wind conditions are best represented by discrete (i.e. deterministic) models. These may be generated using a number of techniques, including situation specific models based on the physics of atmospheric flows (i.e. thunderstorm outflow models[2]) and mathematically optimal worst-case methods.[3] The latter group of techniques seek wind disturbances constrained in a specified manner that maximize a functional of the state of the aircraft. They pose the worst-case concept in a formal mathematical framework while at the same time avoiding the difficult task of modelling complex atmospheric flows. Such techniques have been used in the past to find worst-case wind time histories (e.g. Van der Vaart's method[3]).

For certain types of formulations it is possible to specify the worst-case solutions in terms of the aircraft state.[4] This technique is equivalent to closing the loop on the wind, and appears to be well-suited to flight simulator applications. The level of control difficulty caused by the presence of the variable winds may be adjusted by setting parameters within the wind controller. More importantly, because the human pilot introduces randomness and because the gross features of the control-loop signals are removed by linearizing about a reference equilibrium, the pilot will never see the identical wind profile twice and thus, as in the real world, each encounter will represent a new experience.

In the following the results of a preliminary assessment of the wind controller method for flight simulator applications are presented. Three wind controllers synthesized using one-sided and two-sided (differential game) optimization theory were implemented on a manned three degree-of-freedom fixed-base simulation of a light STOL transport. Data were collected for steep ILS approaches flown by two pilots and an evaluation made of the wind model severity and realism. These results were then compared with others obtained in the presence of two reference wind profiles.

II. Theoretical Background

Two-sided optimization (conflict of interest) problems constrained by systems of differential equations are the subject of differential games.[5] In the broadest sense differential games are a

subset of game theory and are related to it in much the same manner as optimal control theory is to functional minimization. A detailed development of the application of differential games theory to the aircraft controller versus wind conflict of interest problem is given in Ref. 4, and only a few of the key steps in this development will be summarized here.

The conflict of interest between the wind and the aircraft controller is an example of a minimax problem. Such problems may be conceptually viewed as a conflict between intelligent adversaries and may be heuristically defined as follows. The payoff function

$$J[\underline{u}(t), \underline{\eta}(t)] = s[\underline{x}(t_f), t_f] + \int_{t_o}^{t_f} g[\underline{x}(t), \underline{u}(t), \underline{\eta}(t), t]dt \qquad (1)$$

is to be maximized by the disturbance vector $\underline{\eta} \in H$ and minimized by the control vector $\underline{u} \in U$ subject to the differential equations of motion

$$\dot{\underline{x}}(t) = \underline{f}[\underline{x}(t), \underline{u}(t), \underline{\eta}(t), t] \qquad (2a)$$

$$\underline{x}(t_o) = \underline{x}_o \qquad (2b)$$

Here t_o is the initial time and t_f is the final or terminal time. The minimax solution to this problem, if it exists, satisfies the inequalities

$$J(\underline{u}^*, \underline{\eta}) \leq J(\underline{u}^*, \underline{\eta}^*) \leq J(\underline{u}, \underline{\eta}^*) \qquad (3)$$

$$\underline{\eta} \in H \qquad \underline{u} \in U$$

The superscript asterisk denotes the minimax solution. $\underline{u}$ and $\underline{\eta}$ are subject to the constraints implied by the admissible control and disturbance spaces U and H respectively. The payoff function J is defined in a way such that its minimization reflects good controller performance and its maximization reflects poor performance. The payoff function and the admissible control and disturbance spaces must constrain $\underline{u}$ and $\underline{\eta}$ sufficiently to make the problem meaningful, i.e. so that arbitrarily large control and disturbance input energies are not allowed.

With the notable exception of the class of linear quadratic problems, feedback solutions to minimax problems are generally difficult to obtain. For linear quadratic formulations the equations of motion are linear equations of the form*

$$\dot{\underline{x}}(t) = \underline{F}\,\underline{x}(t) + \underline{G}_1\,\underline{u}(t) + \underline{G}_2\,\underline{\eta}(t) \qquad (4a)$$

$$\underline{x}(0) = \underline{x}_o \qquad (4b)$$

and the payoff function is quadratic, i.e.

$$J = \underline{x}^T(t_f)\underline{S}\,\underline{x}(t_f) + \int_o^{t_f} [\underline{x}^T(t)\underline{Q}\,\underline{x}(t) + \underline{u}^T(t)\underline{R}_1\underline{u}(t) + \mu\,\underline{\eta}^T(t)\underline{R}_2\underline{\eta}(t)]dt \qquad (5)$$

$\underline{S}$ and $\underline{Q}$ are positive semidefinite symmetric matrices, $\underline{R}_1$ is a positive definite symmetric matrix, and $\underline{R}_2$ is a negative definite symmetric matrix. The sign definiteness properties of these matrices make the problem meaningful in the minimax context. The positive parameter μ may be adjusted so that the disturbance "energy"

$$E = -\int_o^{t_f} \underline{\eta}^T(t)\underline{R}_2\,\underline{\eta}(t)dt \qquad (6)$$

is within a desired range.

The minimax feedback solution to the linear quadratic problem always exists for large enough values of the constant μ, and is given by

$$\underline{u}^* = -\underline{R}_1^{-1}\,\underline{G}_1^T\,\underline{P}(t)\underline{x}(t) \qquad (7a)$$

$$\underline{\eta}^* = -\frac{1}{\mu}\,\underline{R}_2^{-1}\,\underline{G}_2^T\,\underline{P}(t)\underline{x}(t) \qquad (7b)$$

$\underline{P}(t)$ is the solution to the generalized matrix Riccati equation

$$\dot{\underline{P}}(t) = -\underline{P}(t)\underline{F} - \underline{F}^T\underline{P}(t) + \underline{P}(t)\left[\underline{G}_1\underline{R}_1^{-1}\underline{G}_1^T + \frac{1}{\mu}\,\underline{G}_2\underline{R}_2^{-1}\underline{G}_2^T\right]\underline{P}(t) - \underline{Q} \qquad (8a)$$

$$\underline{P}(t_f) = \underline{S} \qquad (8b)$$

The optimal value of J is given by

$$J^* = \underline{x}_o^T\,\underline{P}(0)\underline{x}_o \qquad (9)$$

Riccati equation (8a) is nearly identical to the Riccati equation that arises in linear quadratic optimal control problems, the difference being the $\frac{1}{\mu}\,\underline{G}_2\,\underline{R}_2^{-1}\,\underline{G}_2^T$ term introduced by the minimax nature of the problem. If one wishes to consider a one-sided maximization, i.e. a minimax aircraft control law is not to be determined, then the worst-case control law continues to be given by (7b) where now the equations of motion (4a) do not contain control inputs $\underline{u}$, the payoff function J of equation (5) does not contain the $\underline{u}^T\,\underline{R}_1\underline{u}$ term, and the Riccati equation (8a) does not contain the $\underline{G}_1\,\underline{R}_1^{-1}\,\underline{G}_1^T$ term. One-sided maximization will be referred to as the direct method.

In the application to flight simulator wind modelling the Riccati equation is solved off-line and the matrix $-\frac{1}{\mu}\,\underline{R}_2^{-1}\,\underline{G}_2^T\,\underline{P}(t)$ is stored. This matrix may then be used in conjunction with (7b) to determine wind inputs in real time as the simulation proceeds. For the purposes of this preliminary study, however, time-invariant control laws based on a value of $\underline{P}$ at a suitably chosen time were used, thereby eliminating the need for large amounts of computer memory for storing a time-varying wind control law.

*For simplicity the linear quadratic problem will be presented as a time-invariant problem. The theory readily extends to time-varying cases.

III. Aircraft Dynamics Model

The aircraft equations of motion are taken to be the longitudinal equations linearized about an equilibrium condition of constant airspeed flight along a rectilinear glide slope in the presence of a constant headwind. The control inputs are elevator angle δ_E and throttle position δ_T. These equations may be written in the matrix form[6]

$$\Delta\dot{\underline{x}} = \underline{A}\Delta\underline{x} + \underline{C}_1\Delta\underline{\delta} + \underline{C}_2\Delta\underline{W} + \underline{C}_3\Delta\dot{\underline{W}} \qquad (10)$$

where

$$\Delta\underline{x}^T = [\Delta u \ \Delta w \ \Delta q \ \Delta\theta \ \Delta x_I \ \Delta h] \qquad (11)$$

$$\Delta\underline{\delta}^T = [\Delta\delta_E \ \Delta\delta_T] \qquad (12)$$

$$\Delta\underline{W}^T = [W_1 - W_{1_e}, \ W_3] \qquad (13)$$

The simulated aircraft is a turbine powered twin-engined light STOL transport of 4500 kg (11000 lb) gross weight. The linearization reference equilibrium conditions used were

$$V_e = 40 \text{ ms}^{-1}$$

$$\gamma_G = 7°$$

$$W_{1_e} = 0$$

The resulting modal characteristics are summarized in Table 1.

The equations of motion (10) were used for the flight simulator dynamic model but had to be rewritten in the form of equation (4a) in order to apply the minimax theory described in the previous section. For the worst-case wind controllers used in this study, a suitable form can be shown to be[4]

$$\Delta\dot{\underline{x}}_{op} = \underline{A}_{op}\Delta\underline{x}_{op} + \underline{C}_{1_{op}}\Delta\dot{\underline{\delta}} + \underline{C}_{3_{op}}\dot{\underline{W}}_{wc} \qquad (14)$$

where

$$\Delta\underline{x}_{op}^T = [\Delta u \ \Delta w \ \Delta q \ \Delta\theta \ \Delta\delta_E \ \Delta\delta_T] \qquad (15)$$

$$\dot{\underline{W}}_{wc}^T = [\dot{W}_{1_{wc}} \ \dot{W}_{3_{wc}}] \qquad (16)$$

Here $\Delta\underline{x}_{op}$, $\Delta\dot{\underline{\delta}}$, $\dot{\underline{W}}_{wc}$, $\underline{A}_{op}$, $\underline{C}_{1_{op}}$, $\underline{C}_{3_{op}}$ have a one-to-one correspondence with $\underline{x}$, $\underline{u}$, $\underline{y}$, $\underline{F}$, $\underline{G}_1$, and $\underline{G}_2$ of equation (4a). The control inputs are the elevator rate $\dot{\delta}_E$ and the throttle rate $\dot{\delta}_T$. This permits the elevator deflection and throttle position to be treated as part of the state vector, allowing wind controller solutions that take pilot control actions into account when determining the worst-case wind inputs. The Δx_I and Δh components of the state vector $\Delta\underline{x}$ of equation (11) are dropped from (15) because they are not treated as being available to the wind controller. $\dot{W}_{1_{wc}}$ and $\dot{W}_{3_{wc}}$ are, respectively, the wind controller contributions to the $\dot{W}_1$ and $\dot{W}_3$ components of the wind vector. Since W_1 and W_3 appear only in the Δx_I and Δh equations (see Ref. 6) they also drop out of (14)

IV. Wind Models

Five wind models were used in this study. Two of these were fixed reference profiles while the remaining three were wind controller models based on the intelligent adversary concept and synthesized using the linear quadratic theory summarized in Section II. No turbulence inputs were introduced in any of the runs.

Model 1 is a constant shear ($0.75 \text{ ms}^{-1}/30$ m) profile representative of light wind shear conditions in which $W_3 = 0$ (see Fig. 1). It was used to obtain baseline data.

Model 2 (Fig. 1) is based on the winds estimated to be present at the time of the JFK accident. The downdraft velocities were reduced by a factor of 0.75 to make them more compatible with the climb capabilities of the simulated STOL transport. This model was used to obtain data representative of flight in extreme wind shear and downdraft conditions.

Models 3, 4 and 5 are the wind controller models. The payoff function weighting matrices [see (5)] under which they were obtained were diagonal and are defined in Ref. 4. The weights were selected so that technically significant deviations in the aircraft state from the reference state were weighted approximately the same and so that the phugoid mode of the aircraft was destabilized (see Table 1). The wind inputs generated by these models were introduced at an altitude of 400m (1300 ft) and were superimposed on the wind profile of Model 1. The latter was done to facilitate comparisons with the baseline data.

For certain situations (e.g. unusually large airspeed deviations), the wind inputs generated by the wind controllers may become unrealistically large. This problem was avoided by specifying wind speed envelopes as shown in Fig. 1. The wind inputs were matched smoothly with these envelopes using an algorithm that was applied continuously to $\dot{W}_1$ and $\dot{W}_3$. This algorithm was based on the ratio of the square of the actual wind speed to the square of the wind envelope speed.

Table 1. Summary of natural mode characteristics with and without wind controllers.

Wind Model	Mode	ζ	ω_n (rad/s)	ω (rad/s)	$T_{1/2}$ or (T_2) (sec)	T (sec)
Open-Loop	Short-Period	0.651	2.56	1.94	0.416	3.24
Open-Loop	Phugoid	0.175	0.298	0.293	13.3	21.4
3	Phugoid	-0.340	0.321	0.302	(6.36)	20.8
4	Phugoid	-0.340	0.453	0.438	(6.00)	14.3
5	Phugoid	-0.157	0.391	0.386	(11.3)	16.3

There is no guarantee that the human pilot will attempt to minimize the same payoff function that the wind controllers are trying to maximize, or that he will even perform optimally. Furthermore, the wind controller optimization did not take into account the presence of the baseline wind profile and the wind speed envelopes. All of these factors make the wind controllers suboptimal, i.e. they are not mathematically worst-case for the simulated flight task. From the practical point of view, however, these differences between the dynamic system for which the wind controllers were optimized and the actual dynamic system to which they were applied were not found to reduce the perversity of the wind controller models to such an extent that it prevented their application to hazardous wind modelling on flight simulators, as will be demonstrated in the following section.

The wind control law for Model 3 is given by

$$\dot{W}_{1_{WC}} = -0.299\Delta u - 0.00722\Delta w + 0.326\Delta q + 1.08\Delta\theta + 0.821\Delta\delta_E - 0.730\Delta\delta_T \quad (17)$$

$$\dot{W}_{3_{WC}} = 0.0136\Delta u - 0.040\Delta w + 0.143\Delta q + 1.46\Delta\theta - 0.834\Delta\delta_E - 0.0344\Delta\delta_T \quad (18)$$

This is based on a steady-state minimax solution* and includes contributions that depend on $\Delta\delta_E$ and $\Delta\delta_T$, i.e. the pilot's control actions are taken directly into account in determining the worst-case wind inputs.

The wind control law for Model 4 does not have W_3 components nor $\Delta\delta_E$, $\Delta\delta_T$ feedback, a formulation that was found to be particularly effective in destabilizing the open-loop phugoid mode of the aircraft. It is based on a direct method worst-case solution with $\underline{P}(t)$ in (7b) replaced by** $\underline{P}(0)$, and is given by

$$\dot{W}_{1_{WC}} = -0.365\Delta u - 0.195\Delta w + 2.19\Delta q + 12.9\Delta\theta \quad (19)$$

Model 5 is of the same type as Model 4, but it is somewhat less destabilizing of the phugoid mode (see Table 1). It is given by

$$\dot{W}_{1_{WC}} = -0.244\Delta u - 0.106\Delta w + 1.20\Delta q + 7.08\Delta\theta \quad (20)$$

V. Description of Experiment

The STOL transport's dynamic characteristics and the wind models were implemented on the UTIAS multi-purpose fixed-base simulation facility. Cockpit instrumentation consisted of an airspeed indicator, an altimeter, an engine power indicator and an electronic attitude indicator with fast-slow and glidepath deviation bugs. The controls available were elevator, throttle and pitch trim. No out-the-window visual cues were presented to the pilots. All simulation variables were updated and sampled at 25 Hz, and recorded on digital magnetic tape.

*Wind control law (7b) with $\underline{P}(t)$ replaced by $\lim_{t\to-\infty} \underline{P}(t)$ (see Ref. 4).

**Because of the presence of conjugate points (see Ref. 4), this model did not have a steady-state solution. The wind control law was arbitrarily chosen to be that control law which results from (7b) with $\underline{P}(t)$ replaced by $\underline{P}(0)$.

The simulated task consisted of intercepting a 7° ILS glidepath from level flight at 460 m (1500 ft) and flying an approach to a 60 m (200 ft) decision height. Pilots were instructed to fly the approach using whatever technique they felt was appropriate, but in all circumstances to continue to the decision height.

Two pilots participated in the study. Subject 1 was an experienced (13000 hours total time) test pilot with over 1000 hours IFR and 400 hours in flight simulators. Subject 2 was a civilian flying instructor (950 hours total time) with 29 hours IFR and 9 hours in flight simulators.

The pilots were briefed on the approach task and then flew familiarization runs with practice variable wind conditions consisting of a W_1 time-referenced sinusoid at the phugoid frequency until they felt comfortable with the simulation. Production runs were flown in blocks of 5 approaches, each block utilizing the same wind model, for a total of 10 runs per wind model per pilot. The order of the blocks was randomized for each pilot. Prior to flying the production runs each pilot was told that he would encounter variable wind conditions but was not informed as to the type and degree of severity of the winds.

Following each approach the pilot was asked to fill out a questionnaire. After each block of five approaches he was interviewed by the experimenters to elicit any further comments regarding the winds encountered and the simulation itself.

VI. Results and Discussion

The data recorded during the production runs included all of the longitudinal state variables, as well as the control variables and wind inputs. Root-mean-square values of these were computed for the runs and the values were analyzed (using t tests) to determine whether significant differences existed with respect to the baseline (Model 1) results and between pilots.

Figures 2 to 4 contain a representative sampling of the wind profiles that were obtained from the three wind controller models. Other than the W_3 inputs of Model 3, the profiles change markedly from run to run and pilot to pilot.

The relatively minor changes observed in the characteristics of the W_3 profiles of Model 3 may be explained by considering in detail the way in which the wind controllers were synthesized and then implemented in the simulation. In the optimal wind controller synthesis process, a linear aircraft dynamics model with $W_{1_e} = 0$ and time-invariant reference equilibrium conditions was used. In the simulation the same aircraft dynamics model was used but the wind inputs generated by the wind controller were superimposed onto the linear wind profile of Model 1. In general the presence of the linear wind results in $\Delta\theta$ and $\Delta\delta_T$ offset components that feed back through the wind controller model to generate W_1 and W_3 offsets. This effect was most prominent for W_3 in Model 3, a consequence of the particular values of the $\Delta\theta$ and $\Delta\delta_T$ gains in that model, and resulted in a significant contribution to W_3 that was unchanged from run to run.

This characteristic can be avoided by deleting

the linear wind or by considering a more sophisticated simulation in which the state variables that produce these offset effects are passed through a suitably defined low pass filter. The perturbation quantities $\Delta\theta$, $\Delta\dot{h}_T$, and so forth on which the wind controller operates can then be defined with respect to the filtered quantities rather than with respect to their reference equilibrium values, thus eliminating the unwanted offsets in the wind controller output.

The wind variability was measured by the quantity

$$S_{ws} = \int_0^{t_f} [\dot{W}_1^2(t) + \dot{W}_3^2(t)]dt \qquad (21)$$

This quantity is analogous to the integral E of equation (6). t_f was in the range $90s \leq t_f \leq 105s$ for all of the runs, but showed less variation for runs involving a particular wind model. S_{ws} depends on the time rate of change of the wind velocity as seen by the aircraft, and will thus change from run to run even for runs in the presence of the reference profiles of Models 1 and 2.

As well as producing different wind inputs from run to run, the wind controller models also showed considerable S_{ws} variation both among the wind models and between pilots. This is apparent from Table 2. The quantity $\sigma_{S_{ws}}/\mu_{S_{ws}}$ is generally larger for the wind controller models than for the fixed profiles of Models 1 and 2. Also, the S_{ws} values for Models 2 to 5 are generally larger for Pilot 1 than for Pilot 2.

For Models 2 to 4 these differences in $\mu_{S_{ws}}$ between pilots were shown to be significant at the 5% confidence level (see Table 5). This suggests that some fundamental differences existed in the control strategies adopted by the two pilots. Since the wind controllers determine wind inputs based on a subset of the aircraft state and control inputs, certain control strategies will therefore produce different wind characteristics. This is also suggested by a comment, repeated several times by Pilot 1, that he concentrated on obtaining good glidepath tracking and was willing to tolerate a "few knots" of airspeed deviation. Since none of the wind controller models contain glidepath deviation in their wind generation algorithms while they all contain airspeed deviation, they would tend to produce less severe winds for pilots who adopted a tighter airspeed tracking strategy. Conversely, a pilot who adopts a tight glidepath tracking strategy at the expense of airspeed tracking might ultimately find himself in a divergent situation where both airspeed and glidepath tracking performance are degraded markedly. Such an effect was seen for Pilot 1 during many of his approaches involving Model 4 where the wind inputs would oscillate from one side of the limiter envelope to the other.

The hazard posed by the wind controller models was evaluated using a number of techniques, including the following:

1. Comparison of S_{ws} values obtained for the wind controller models with those obtained for the fixed profile of Model 2, which is known to be extremely hazardous to aircraft.

2. Categorization of the wind shear encountered for a given W_1 profile in 30m altitude increments according to the ICAO interim classification[7] of Table 3.

3. Subjective evaluation through the pilots' response to the questionnaire.

Each of these will be discussed briefly in the following.

In general the S_{ws} values that were obtained for the wind controller models were smaller than those obtained for Model 2. These generally smaller values do not necessarily imply that the winds generated by these models are not hazardous. S_{ws} represents an average value of wind variability; large shears may still have existed even in runs where S_{ws} was not large. As an example of this the second W_1 wind profile of Fig. 2 has been categorized as to shear strength according to the ICAO classification of Table 3. From Fig. 5 it is seen that despite the relatively small value of S_{ws}, there are a number of strong and severe shear encounters.

The pilots' assessment of the wind hazard through the questionnaire also produced some interesting differences between them. For all of the wind models, as compared to Pilot 2, Pilot 1 generally rated the winds encountered as being more hazardous, the flight task as being more difficult, and the aircraft controllability as being more marginal. As an example of this trend Fig. 6 shows the number of go-arounds that each pilot felt should have been executed if this had been allowed. Pilot 1's greater number of go-arounds is compatible with the larger S_{ws} values that he obtained for many of the wind controller runs. We note, however, that even for Model 5, for which the S_{ws} mean and standard deviations for the two pilots were similar (see Table 2), Pilot 1 would still have executed a greater number of go-arounds.

Both pilots had difficulty determining the

Table 2. S_{ws} summary for all wind models.

	Pilot 1			Pilot 2		
Wind Model	$\mu_{S_{ws}}$ ($m^2 s^{-3}$)	$\sigma_{S_{ws}}$ ($m^2 s^{-3}$)	$\frac{\sigma_{S_{ws}}}{\mu_{S_{ws}}}$	$\mu_{S_{ws}}$ ($m^2 s^{-3}$)	$\sigma_{S_{ws}}$ ($m^2 s^{-3}$)	$\frac{\sigma_{S_{ws}}}{\mu_{S_{ws}}}$
1	0.9	0.0	0.0	0.9	0.0	0.0
2	127.7	15.0	0.12	104.7	20.8	0.20
3	10.3	4.8	0.47	5.1	3.0	0.59
4	111.8	43.2	0.39	12.4	5.2	0.42
5	5.8	2.8	0.48	5.2	2.1	0.40

Table 3. ICAO wind shear classification

Category	Vertical Shear Magnitude ($ms^{-1}/30m$)
Light	0 - 2.5
Moderate	2.5 - 4.5
Strong	4.5 - 6.0
Severe	> 6

type of wind disturbance (i.e. W_1, W_3 or both) which they had encountered on a given approach. Pilot 1 commented that he would have been able to make a better assessment of the type of wind inputs if inertial acceleration cues had been available.

Both pilots generally found the wind controller models useful for training purposes, although for some of the runs they found the winds that they had encountered to be unrealistically severe. We note that the fixed profile of Model 2, which is based on an estimate of the severe variable wind conditions that existed during the JFK incident, was also rated as being unrealistic for many of the runs.

t tests were employed to examine the differences between the results produced by wind Models 2 to 5 and the baseline (Model 1) and the differences between the two pilots. These tests were performed on the mean value of the RMS response averaged over the 10 runs per subject for each wind model. The results are summarized in Tables 4 and 5. From Table 4 it can be seen that with few exceptions significantly larger values of the RMS levels were obtained for wind Models 2 to 5 than for Model 1. The least increase in RMS level was found for throttle rate activity. From Table 5 it can be seen that Pilot 1 produced significantly larger RMS values for most variables than Pilot 2, as is indicated by the positive values for the t statistic.

VII. Summary and Recommendations for Future Work

Hazardous wind generation based on the intelligent adversary concept and implemented as wind controller models has been evaluated on a fixed-base flight simulator. The major advantage over pre-recorded deterministic models is that this technique produces wind inputs that change from run to run and pilot to pilot. These low frequency wind inputs tend to excite the more weakly damped low frequency modes of the pilot-aircraft system, and thus are useful for hazard definition.[3,8] The wind controller models were generally considered by the pilots to produce disturbances that are useful for training purposes. Furthermore, some of the results suggest that the wind controller models may be formulated in a manner that favours certain pilot control strategies over others. This characteristic may ultimately prove to be useful as a training tool.

It is recommended that future work in this area should be undertaken to study

1) the implementation and assessment of the wind controller technique on more sophisticated six degree-of-freedom moving base flight simulators,

2) the level of pilot adaptation to wind controller models as compared with the learning effects associated with pre-recorded deterministic models, and

3) the development of real time techniques for controlling the level of wind variability and pilot workload for a given wind controller model.

Table 4. Statistics summary:[a] Rise in mean RMS values for wind Models 2-5 over baseline (Model 1) runs

Wind Model	Pilot 1					Pilot 2				
	$\hat{\Delta u}$	$\hat{\Delta d}$	$\mu_{S_{ws}}$	$\hat{\delta}_T$	$\hat{\delta}_E$	$\hat{\Delta u}$	$\hat{\Delta d}$	$\mu_{S_{ws}}$	$\hat{\delta}_T$	$\hat{\delta}_E$
2	21.** † (4)	14.5** (4)	18.9** (4)	1.6 (8)	7.8** (5)	4.** (9)	5.87** (9)	15.7** (9)	2.43** (13)	8.5** (10)
3	8.** (10)	3.78** (12)	6.3** (9)	0.36 (11)	4.6** (13)	3.** (13)	3.0** (17)	4.67** (9)	0.57 (12)	4.** (9)
4	10.7** (9)	4.35** (9)	8.1** (9)	2.0* (17)	7.5** (11)	7.** (10)	1. (15)	7.18** (9)	1. (10)	12.** (15)
5	4.** (13)	.83 (13)	5.4** (9)	-0.82 (10)	0.67 (17)	5.** (13)	-0.3 (16)	6.1** (9)	1.14 (15)	11.** (13)

[a]Based on one-tailed t-tests for two populations having different and unknown variances, testing H_0: the mean RMS values for wind model runs are equal to those for the baseline runs.

*Significant at the 0.05 level.

**Significant at the 0.025 level.

† t value
(degrees of freedom).

Table 5. Statistics summary:[a] Inter-pilot comparison of mean RMS values for wind models

Wind Model	$\hat{\Delta u}$	$\hat{\Delta d}$	$\mu_{S_{ws}}$	$\hat{\delta}_T$	$\hat{\delta}_E$
1	2.† (18)	3.8** (17)	b	0.3 (14)	9.1** (10)
2	3.5** (13)	5.7** (12)	2.4* (10)	1.1 (4)	8.9** (5)
3	4.** (13)	4.6** (10)	2.9* (15)	0.8 (16)	6.1** (15)
4	8.3** (12)	4.9** (10)	7.2** (9)	2.7* (9)	8.9** (9)
5	0.0 (18)	3.7** (11)	0.5 (16)	-2.6* (14)	5.3** (15)

[a] Based on two-tailed t-statistics for two populations having different and unknown variances, testing Ho: the mean RMS values for each wind model are the same for each pilot.

[b] Indeterminate because t = 0/0.

*Significant at the 0.05 level.

**Significant at the 0.01 level.

† t value
(degrees of freedom)

Acknowledgement

This work was supported by grants from the Natural Sciences and Engineering Research Council of Canada and the Transportation Development Agency of Transport Canada. The authors also gratefully acknowledge the willing participation of the two pilots provided by de Havilland Aircraft of Canada Ltd.

References

1. Frost, Walter, and Camp, Dennis W., Wind Shear Modeling for Aircraft Hazard Definition, Interim Report, FAA-RD-77-36, March 1977.

2. Williamson, G. G., Lewellen, W. S., and Teske, M. E., Model Predictions of Wind and Turbulence Profiles Associated with an Ensemble of Aircraft Accidents, NASA CR-2884, July 1977.

3. Van der Vaart, J. C., Worst-Case Wind Time Histories Causing Largest Deviations from a Desired Flight Path, Report LR-267, Delft University of Technology, April 1978.

4. Markov, A. B., The Landing Approach in Variable Winds: Curved Glidepath Geometries and Worst-Case Wind Modeling, Ph.D. Thesis, University of Toronto, Institute for Aerospace Studies, March 1981.

5. Isaacs, R., Differential Games, John Wiley & Sons, 1965.

6. Reid, L. D., Markov, A. B., and Graf, W. O., The Application of Techniques for Predicting STOL Aircraft Response to Wind Shear and Turbulence During the Landing Approach, UTIAS Report No. 215, June 1977.

7. Fichtl, George H., "Wind Shear Near the Ground and Aircraft Operations", Journal of Aircraft, Vol. 9, No. 11, November 1972.

8. Turkel, Barry S., and Frost, Walter, Pilot-Aircraft System Response to Wind Shear, NASA CR-3342, November 1980.

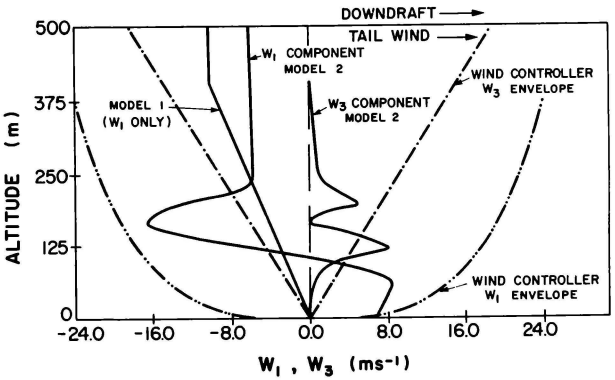


Fig. 1 Reference wind profiles and wind speed envelopes.

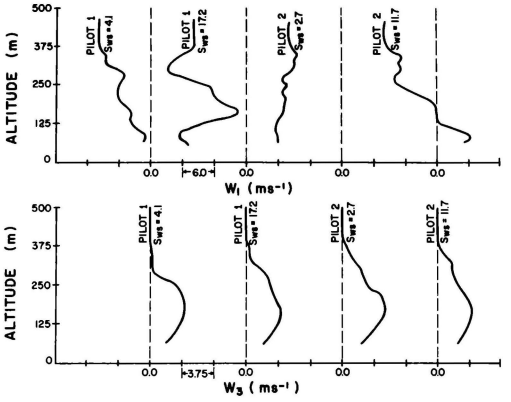


Fig. 2 Wind Model 3.

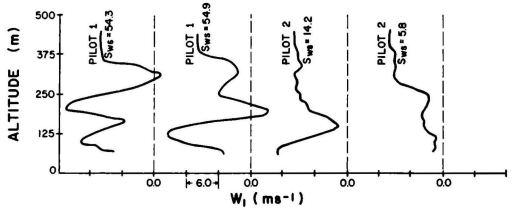


Fig. 3 Wind Model 4.

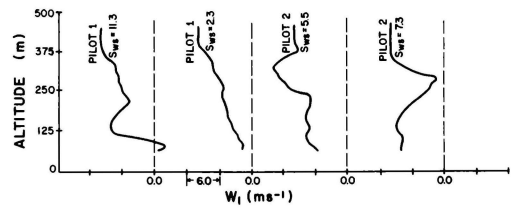


Fig. 4 Wind Model 5.

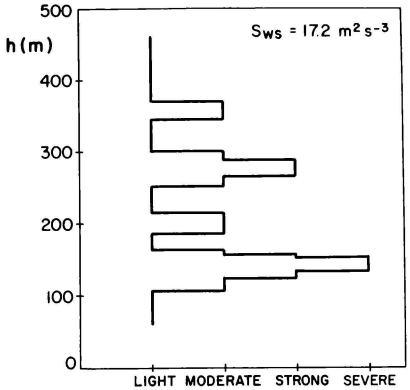


Fig. 5 Shear classification of second W_1 profile of Fig. 3.

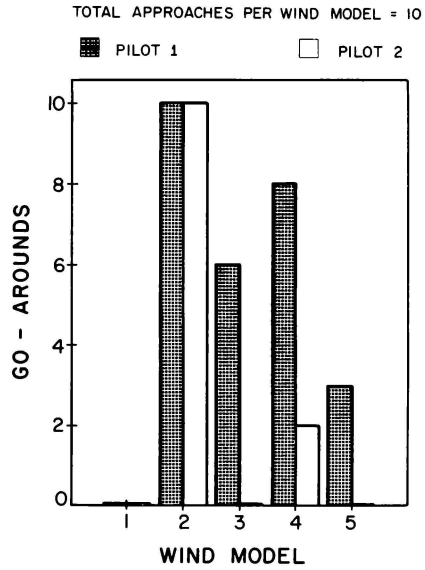


Fig. 6 Summary of number of approaches for which go-arounds would have been executed.

A SIMULATION APPROACH TO MIL-STD-1553 MULTIPLEX BUS INTERFACING

Robert J. Lawson*
Jack Murray**
Gould, Inc.
Simulation Systems Division
Melville, New York 11747

Abstract

A technique has been developed for interfacing a 32 bit minicomputer to the MIL-STD-1553 Avionics Multiplex Bus (Mux Bus) in the F/A-18 Part Task Trainer. The capability of access to the Mux Bus through a minicomputer provides the means of emulating any aircraft system the mission computer interfaces to in the aircraft. The capability of emulating the mission computer also exists for the purpose of stimulating real aircraft systems. The technique for recognizing bus requests for systems data required for simulation and responding to these requests within the timing constraints of 1553 is described in this paper along with the details of bus operation specified by 1553.

Introduction

The bulk of cockpit simulation is changing from one of stimulating aircraft systems using simulated hardware and software, to the task of using complex systems right out of the aircraft for the trainer and making these systems feel as though they never left the aircraft. The advantages are many for this technique. As aircraft systems are modified in the field, it is easy enough to replace the system in the trainer. Also, training integrity is maintained with little or no impact to the simulator.

Most aircraft systems transmit and receive data under control of one or more aircraft computers. These computers are being transplanted to the simulator with some of the systems that particular computer controls in the aircraft. The operational software for that computer is also used, unmodified, in the simulator. The job for the simulation hardware and software, then, is to make the computer think all of its flight systems are attached and operating. During the course of the training exercise, it is necessary to provide the flight computer with simulated systems data, such as radar modes, target position, own aircraft position, etc., that affect the training exercise.

*Electrical Engineer, Lead E.E. F/A-18 PTT
**Sr. Software Engineer, Lead S.E. F/A-18 PTT

To provide the aircraft computer in the F/A-18 trainer (this computer is referred to as the mission computer) with systems data the 1553 bus interface is used. The use of the 1553 bus greatly reduces interconnect wiring of the computer to its associated systems by utilizing a twisted, shielded pair cable as opposed to many discrete, parallel data lines with associated control signal lines. In the simulation, therefore, it becomes very easy to connect the simulation computer to the mission computer via a single twisted, shielded pair cable and now become a part of the computers' bus system. Responding to mission computer commands to aircraft systems being simulated will now occur via this interconnect to the simulation computer.

1553 Mux Bus Word

The hardware interface used to interface the aircraft computer to its flight systems is the MIL-STD-1553 Avionics Multiplex Bus or, as referred to throughout the rest of this paper, the Mux Bus.

The bus is an asynchronous serial data bus operating at 1 MHz and using bi-polar voltage levels of plus and minus V volts for noise immunity. (The F/A-18 bus has a V of ≈5 volts). The serial data is Manchester encoded to further enchance low error data transmission and the transmission line is a differential, transformer coupled bus terminated at both receiver and transmitter at the characteristic impedance of the line. Figure 1 illustrates the Mux Bus transmission line.

The serial data on the bus is a 16 bit word format. Each word is preceded by a 3 microsecond, invalid Manchester sync code. Since the data on the bus is sent asynchronously, this sync time is used to set-up the hardware to receive data.

Sync is also used, as per 1553, to define the type of word coming across. A high to low transition of sync defines both command and status words while a low to high transition defines data words. Since sync is 3 microseconds wide total, the transition (low to high or high to low) occurs at 1.5 microseconds.

The information following sync is a 16 bit Manchester encoded word, either a data, command, or status word. In keeping with the 1 MHz bus rate, each bit is 1 microsecond wide. As per the Manchester code, a low to high transition would indicate a logic 0 while a high to low transition represents a logic 1. As shown in Figure 2, this encoding is in sync to a 1 MHz clock.

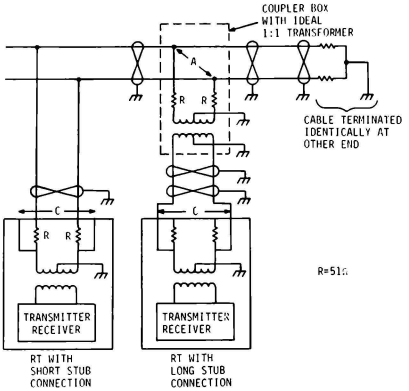


Figure 1. Mux Bus Transmission Line

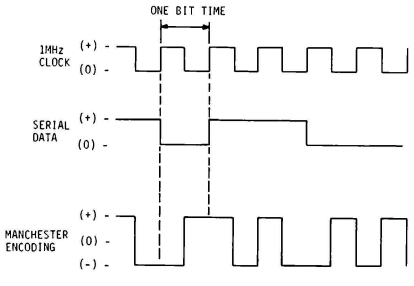


Figure 2. Manchester Encoding of Mux Bus

The last part of the word needed in a transmission is the parity bit. This is again, a valid 1 microsecond Manchester coded bit and as defined by 1553, parity is odd.

To summarize, a Mux Bus word is comprised of sync, 16 bit word and parity. With no time gaps allowed between these entities, a Mux Bus word takes a total of 20 microseconds to be transmitted or received.

Types of Mux Bus Transmissions

There are four types of transmission that can occur on the Mux Bus. They are:

(1) Bus Controller (BC) to Remote Terminal (RT)

(2) RT to BC

(3) RT to RT

(4) BC to RT Mode Commands

As defined in 1553, the bus controller is the processor which controls bus traffic. A remote terminal is a device connected to the bus which responds to BC commands. An RT can be a BC or vice versa but only one BC is allowed control of the bus for any given transmission. As per 1553B, the BC function can be passed from one controller to another. The controller in command of the bus at the time requests that another controller take command of the bus. The second controller then responds to the first either that he will or will not take control. Should he not accept control, the first controller must remain in command of the bus. The explanations for the four types of transmissions are as follows:

(1) BC to RT

The BC issues a command word on the bus. In the 16 bit word is information for RT address, a transmit/receive bit, a subaddress field for addressing subsystems within a given RT, and a word count field. All RT's on the bus monitor this command word. The RT who recognizes his address in the command word then "opens up" his port to receive data. The data immediately follows the command word with no time gap. The RT, after the last word is received, then has 2 to 5 microseconds to respond to the BC with a status word as per 1553A. 1553B has relaxed this time gap to 4 to 12 microseconds. The status word tells the BC the terminal status is coming from, a bit for indicating if the RT found any errors (parity or Manchester code) in the BC's transmission, a field for status codes used by the RT for special purpose bus functions, and a terminal flag bit used to indicate to the BC that it should examine the status code field.

(2) RT to BC

The BC again issued a command word to an RT to transmit data. 2 to 5 microseconds later, (or 4 to 12 microseconds) that RT sends back a status word followed immediately by the number of data words requested by the BC.

(3) RT to RT (not currently used in the F/A-18 Mux Bus).

The BC now issues two command words, one telling RT A to receive X number of words, another command telling RT B to transmit X number of words. RT B will then transmit a status word (2 to 5 microseconds after receipt of the command word) followed by X number of data words. 2 to 5 microseconds later, RT A will send a status word.

(4) BC to RT Mode Command

The BC issues a command word to an RT and that RT returns a status word. Figures 3 and 4 summarize the transmission formats.

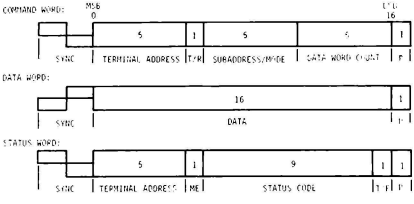


Figure 3. Mux Bus Word Formats

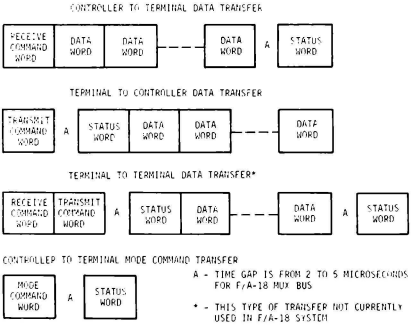


Figure 4. Mux Bus Message Formats

Now that we have laid out the design criteria for the interfacing to the Mux Bus, let us look at this interface from a simulation point of view and see how we can stimulate the mission computer with responses from simulated aircraft systems.

Responding to the Aircraft Computer

Since we are talking about a multiplex data bus, one need only couple onto the bus to become a subscriber. In the F/A-18 aircraft simulator we are using the aircraft computer intact with operational software, unmodified. The F/A-18 software utilizes 1553A which does not allow for dynamic bus control. So, the simulation computer must act as a slave to the aircraft computer; as a remote terminal. The role of the simulation computer will be to respond to bus controller commands to remote terminals that are not physically in the simulator but nonetheless must provide data to the BC. Even in the case where certain aircraft systems are not required for simulation, it is possible the BC will require a response if it quiries that system. In this case, the simulation computer would provide some fixed "dummy" response that would satisfy the BC and yet not affect simulation. The requirement to respond, in the first place, would be a function of the operational flight program.

The role the simulation computer must play now is to interface to the Mux Bus, decode command words being outputted by the bus controller, and determine if the simulation computer should respond or not. Criteria to respond by the simulation computer would be if the BC issued command words to aircraft systems that are not physically present in the simulator. Such a case might be the radar system.

Since there can be no radar tracking in a simulator, none of the hardware associated with radar need be present in the simulator. It is easier to allow the simulation computer to develop math models and output data that make the flight computer think a radar system exists.

Timing Requirements for Bus Response

Clearly, the simulation computer has to satisfy certain timing requirements of the Mux Bus to make the response to the bus controller valid. It would certainly be a large burden on a given Input/Output (I/O) driver of a computer to handle such timing. It becomes more convenient to utilize a dedicated I/O channel in the simulation computer that will only interrupt computer memory when data is required. The I/O driver must also be compatible to the electrical requirements of the 1553 Mux Bus.

The simulation computer must take in and decode all command words from the bus controller. It must determine if the RT address contained in the command word needs a simulation response. In the case of a BC to RT transfer, the data will immediately follow the command word.

In the case of RT to BC transfers, the simulation computer again decodes all command words. In this case, however, the BC does not expect the status word response for 2 to 5 microseconds. This is plenty of time for the simulation computer to determine if it should respond or not.

In the case of RT to RT transferring, several different types of transfers can occur in the simulation environment. They are:

(1) Between two RT's that are physically present in the simulator. In this case the simulation computer would have no role.

(2) Between an RT physically present and an RT that is simulated. Here the simulation computer responds to an RT in the simulator.

(3) Between two RT's that are not physically present but are simulated. Here the simulation computer need only respond on the Mux Bus with the two status words for each of the two RT's. The timing requirement for the simulation computer response, here, is basically the same as an RT to BC transfer.

F/A-18 Mux Bus Configuration

As stated earlier, the F/A-18 Part Task Trainer is utilizing the two mission computers with operational flight software from the actual aircraft. A 1553 bus is utilized between the two mission computers for the purpose of monitoring each other's performance. The intercomputer communications serve no purpose in the simulation environment and therefore the simulation computer will not act on this bus.

The mission computer has two Mux Bus I/O ports. Each bus (designated here as bus 1 and bus 2) is capable of independent 1553 bus communications. It is, therefore, evident that simultaneous bus traffic can occur on bus 1 and bus 2, and the Mux Bus interface must be capable of handling such a situation. Within a main bus; i.e. bus 1, is a pair of 1553 transmission lines acting in a redundant mode (here designated bus 1x and bus 1y or bus 2x and bus 2y). The F/A-18 mission computer does use the redundant pair for bus transmissions, but bus transmission cannot occur simultaneously on bus 1x and 1y as per 1553.

Therefore, the worst case bus transmission that can occur is simultaneous bus transmission on one line of each of the main buses (1 and 2).

Another bus consideration is that, as per 1553, all remote terminals must respond on the bus that the command word was issued on. For simulation purposes, therefore, all four buses must be monitored for command words, and the simulation interface must be capable of responding on all four buses, with the added consideration of being able to monitor simultaneous commands on the four possible combinations of bus transmission (1x and 2x, 1x and 2y, 1y and 2x, 1y and 2y).

Simulation Computer to Mux Bus Interface

For the F/A-18 Part Task Trainer being developed by Gould, we are using a Systems Engineering Laboratories (SEL) 32/77 32 bit minicomputer with 600 nanosecond memory access for simulation. The dedicated I/O processor being used for Mux Bus interfacing is a SEL Regional Processor Unit (RPU). The RPU is a 32 bit microprocessor with 2K of Read Only Memory (ROM) for processor program storage and 4K of high speed Random Access Memory (RAM) for scratch pad memory. The RPU utilizes the SEL 150 nanosecond system clock to develop timing and enable execution of one RPU microinstruction to happen within 150 nanoseconds.

Another feature of the RPU is the Device Interface (DI) board. This is an area set aside for user interface hardware between the RPU and the outside world. The DI board is where the hardware will reside for electrical conversion of RPU data to the 1553 Mux Bus format.

To effect control of the RPU microcode from the DI area, test signals from the user hardware are utilized. The RPU will address a bank of multiplexers on the DI to inquire the state of a given signal and the multiplexer will respond with that state. The RPU can now make logical branch decisions in its microcode depending on the response of the signal it quiered. Up to thirty-two (32) test structures can be addressed by the RPU on an individual basis.

The RPU can effect control over the DI area utilizing order structures. Here, the RPU can command in the microcode the setting, resetting, or pulsing of a selected line to the DI area for control of the 1553 processing hardware. There are a total of 24 order structures to control the DI hardware. All electrical levels up to the 1553 output are TTL logic levels of 0 to 5 volts.

Data traffic between the RPU and DI area is handled in 16 bit parallel format. Two 16 bit registers under control of the RPU microcode are available for data output to the DI area. One 16 bit register is available for inputting data to the RPU.

RPU Controller Firmware

The Host Computer/Trainer Mux Bus Interface Hardware is controlled by firmware that executes in the RPU Controller. This firmware is interrupt driven and responds to host requests for data transfer and control and trainer Mux Bus requests for simulated data. The firmware initiates host memory transfers to satisfy data requirements of the trainer Mux Bus requests. The firmware will be discussed in two parts. The host interface as used for initialization and debug of the controller firmware and the real time responses to trainer Mux Bus requests.

Host Interface

The Controller Firmware responds to host requests for data transfers between its memory and the controller via interrupts. The request is responded to with standard data or control signals that are required by the host computer, such as ready, acknowledge, etc. The request is then decoded and acted upon. The allowable requests include data transfers between host memory and controller RAM, start execution commands, controller register information and controller status. These requests are for setup and debug purposes and are not used during real time operation. The approach is to setup, in the controller RAM, any data that will be required for real time operation, thus relieving the real time firmware of this responsibility and, therefore, improving its response time. The Mux Bus data structure, however, is too large to be stored in RAM and thus address tables are stored instead. These tables are used to obtain the host address of the particular remote terminal data structure. Figure 5 indicates the processing done to obtain Mux Bus data for a given Remote Terminal. The transfer of data between the host and the controller for real time operation is done on a cycle steal Direct Memory Access (DMA) and thus transparent to the software running in the host and processing this data.

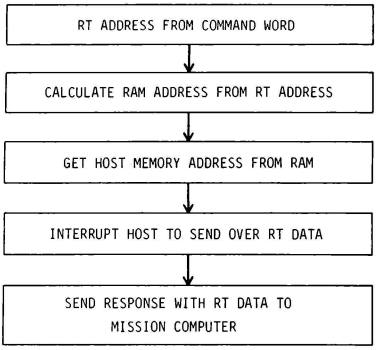


Figure 5. Remote Terminal (RT) Address Calculation

Trainer Mux Bus Interface

When there is activity on the Trainer Mux Bus, the hardware (Mux Bus converter/RPU controller) causes an interrupt into the real time firmware. A binary search of the possible interrupts is done, and the interrupt service routine for the appropriate bus and service requested is entered. The occurrance of an interrupt signifies that a Mux Bus word has been received by the Mux Bus converter or the Mux Bus converter is ready for the next transmit word. This received word is from one of the buses (two buses in this case) and can be either command or data. The data is accompanied by parity error, and sync type tags that further define the type of word received. The firmware checks for errors and resolves word type.

The possible interrupts sent to the controller from the Mux Bus hardware are Mux Bus command word received, Mux Bus data word received, or ready for next Mux Bus data transmission. The following is a brief description of each.

(1) Command Word Received

(a) Any current activity on the affected bus is terminated.

(b) The remote terminal address, message number, and word count is obtained from the received word.

(c) A table look-up is performed to determine if this remote terminal is not simulated or if further examination indicates the message is not supported, no processing is done, and the interrupt service routine is exited, otherwise processing continues. Using tables previously loaded into the controllers RAM, the host address of the remote terminal data area is found.

(d) The mode is set for the appropriate bus (Transmit/Receive).

(e) If the command is to send data, a host read request is queued and a status word is sent to the Mission Computer. The requested number of data words will then follow as outlined in process type 3.

(2) Mux Bus Data Received

(a) If the bus is in the receive mode, the following processing is done.

(b) The data is stored in controller RAM for subsequent transmission to host memory. Since the storing of data in host memory is not time critical, it will be done in the background.

(c) Received data word count is updated.

(d) If the word received is the last data word of the current request and all is well, a status word is sent to the mission computer to inform it of successful data transmission.

(3) Mux Bus Ready for Next Transmit Word

(a) If transmission is complete, the bus is set inactive, and the interrupt service routine is exited. Otherwise processing is as follows.

(b) Data previously stored in controller RAM is sent to Mux Bus converter for transmission to trainer Mux Bus.

(c) Word count and RAM address registers are updated.

Figure 6 is an overall block diagram of the RPU system.

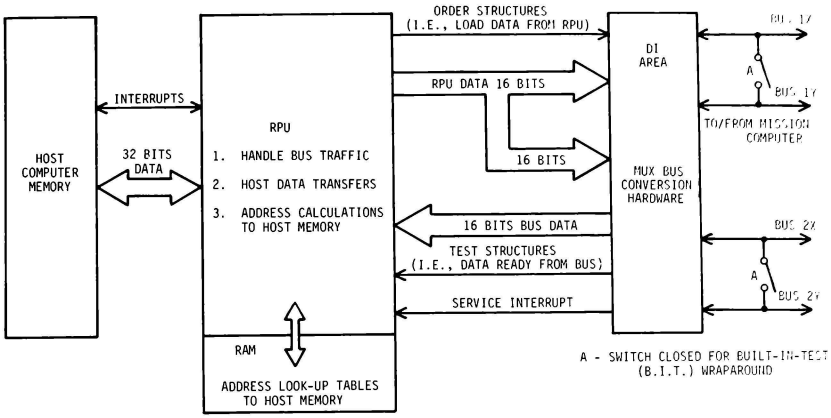


Figure 6. RPU/Mux Bus Block Diagram

Host to Firmware Data Transfers

Address calculation for the area in host memory containing data for a particular RT is accomplished by using tables that are indexed by remote terminal number and message number. The host address is thus obtained by table look-up of tables in RAM. This look-up can be accomplished rapidly and requires no host interaction. Once the address is obtained, the host data transfers can be done to the RPU controller. Reads from the host memory are requested by the firmware via interrupt. When the host data is ready, the firmware is interrupted to receive data. This interrupt approach eliminates any waits and thus increases speed.

Built-In-Test (BIT) of the Mux Bus Hardware

As a diagnostic tool for troubleshooting the RPU/Mux Bus hardware interface, a BIT philosophy has been adopted. By closing a set of microswitches in the Mux Bus hardware, connection of the redundant bus pairs in each of the main buses is made. The Mux Bus hardware can now be checked in a wrap-around mode. The only constraint is that the mission computer is in a halted state so that its bus traffic is eliminated. Via the host computer, the diagnostic routine resident in the controller firmware can now be executed. The controller will send out a known pattern of data words containing different bit patterns (alternate ones and zeroes (checkerboard), reverse checkerboard, etc.). This data is then electrically wrapped back into the controller and can then be checked for faithful reproduction. This test also verifies the functioning of bus interrupts to the controller and controller interrupts to the bus hardware.

Bus Controller Simulation

For the F/A-18 trainer, the simulation involved responding to the mission computer commands to aircraft systems (remote terminals). It can be seen that with modification to the RPU firmware, the simulation computer could be made to act as a bus controller as well as respond as a remote terminal. The application here might be to have the operational flight software compatible to download into the simulation computer. The simulation computer could then handle all requests from this software for flight systems data internally without trafficing data on the Mux Bus. Any systems physically connected in the simulation to the bus would be operated in the command-response mode of the Mux Bus by the simulation computer acting as bus controller. The advantage here would be to eliminate using the actual flight computers and still be able to keep up with flight software changes by having the download capability of the flight software into the simulation computer.

Summary

The requirement to interface to the 1553 Mux Bus for the purpose of simulation was made by the need for an F/A-18 trainer. The trainer utilizes the mission computer with operational flight software and some flight displays that act as remote terminals to the mission computer. The purpose of the

simulation computer is to stimulate the mission computer with simulated flight data for those systems necessary for training. The simulation computer, therefore, must interface to the mission computers' 1553 interface.

The simulation computer interfaces to the Mux Bus via the Regional Processor Unit (RPU). The RPU is a dedicated I/O controller that handles bus traffic via interrupt driven firmware. There is a hardware area on the RPU that handles Mux Bus format conversions and interrupts the firmware with Mux Bus status.

Response to various remote terminals the mission computer is commanding is handled by address look-up tables in RPU RAM. These tables point to locations in host memory for fetching or storing R.T. data. Data transfer from host memory are handled on a DMA basis, therefore, not interrupting host processing. Address look-up calculations are done by RPU firmware for quick transfers. All four Mux Buses of the mission computer are monitored by the mission computer, and response on the bus is done within the timing constraints of 1553.

References

1. Military Standard, Aircraft Internal Time Division Command/Response Multiplex Data Bus, MIL-STD-1553A and MIL-STD-1553B, 21 September 1978, Department of Defense, Washington, DC 20360.

2. F-18 Avionics Multiplex Design Specification, Report Number MDC A3818, Revision date 1 October 1976, Revision B, McDonnell Aircraft Company, Box 516, St. Louis, Missouri 63166.

3. Horattas, Chris G.: "Multiplex Data Bus Applications in Aircraft Simulators". Goodyear Aerospace Corporation, Defense Systems Division, 1210 Massilion Road, Ackron, Ohio 44315.

4. Brickner, David R.: "Military Multiplex Standard Defines Versatile Serial Bus". Computer Design, December 1979, Sperry Flight Systems, Phoenix, Arizona.

5. User Reference and Design Manual, Regional Processor Unit, Models 9144/9145, Systems Engineering Laboratories, Inc., March 1979.

6. "Binary-to-Manchester Encoders", Technical Support Package, NASA Tech Briefs, Summery 1979, Vol. 4, No. 2 MSC-16546, Lyndon B. Johnson Space Center, Houston, Texas 77058.

PILOT/VEHICLE MODEL ANALYSIS OF VISUAL AND MOTION CUE REQUIREMENTS IN FLIGHT SIMULATION

R. Lancraft, G. Zacharias, S. Baron
Bolt Beranek and Newman Inc.
Cambridge, Massachusetts

ACKNOWLEDGEMENT

The work reported here was performed under NASA Ames Research Center Contract Number NAS2-10145. The authors wish to thank Mr. Frank Crane, technical monitor for NASA, for his assistance throughout the course of the program.

Abstract

The optimal control model for pilot/vehicle analysis is used to explore the effects of a CGI visual system and motion system dynamics on helicopter hover simulation fidelity. This is accomplished by expanding the perceptual aspects of the model to include motion sensing and by relating CGI parameters to information processing parameters of the model. Simulator fidelity is examined by comparing predicted performance and workload for flight with that predicted for various simulator configuration.

The results of the analysis suggest that simulator deficiencies or a reasonable nature (by current standards) can result in substantial performance and/or workload infidelity. Both CGI and motin system effects are significant for this task. There is also a distinct interaction between the two sources of pilot cues. In particular, the presence of motion reduces the sensitivity to CGI limitations.

Introduction

As flight control and management tasks become more complex so, too, do the simulators used to investigate these tasks. The designers of simulations are confronted with difficult choices between requirements for simulation fidelity and the needs for cost-effective methods of simulation. The latter demands have resulted in a trend toward the use of digital equipment in simulation both in modeling the vehicle and in generating visual cues (CGI systems) for the pilot of the simulator. These digital simulations can have characteristics that are significantly different from those desired. In particular, unwanted delays frequently are present in such a simulation. When motion cues are also needed, the problems can be aggravated further both by delays in generating the cues (even with analog hardware) and by the potential lack of correlation between visual and motion cues. The significance of these problems has been amply demonstrated in recent studies (Gum and Albery,[1] Queijo and Riley).[2]

In this paper, the optimal control model (OCM) for pilot/vehicle analysis is used to investigate the possible closed-loop consequences in a helicopter hover task of simulator limitations associated with a computer generated image (CGI) visual system and a six degree-of-freedom motion platform (VMS). The potential effects of CGI and VMS system characteristics on closed-loop hover performance and pilot workload are predicted and compared with model predictions for actual flight. To accomplish this, the basic OCM is elaborated to include sensory perception of both CGI-generated visual cues and VMS-generated motion cues.

Model Implementation

In this section, we describe how the simulated hover task is modelled in the context of the Optimal Control Model (OCM) of the pilot.

The basic modelling approach is to define simulator hardware and human sensory dynamics, where appropriate, and to form a mapping between simulator and human perceptual limitations and established OCM parameters. This modified OCM will then be used to predict closed-loop performance as a function of simulator parameters. Inasmuch as the basic OCM has been documented extensively,[3,4] the discussion of the model will be brief, and will emphasize features that are of special relevance to this study.

Figure 1 presents in block diagramatical form the structure of an "expanded" OCM designed to focus on perceptual issues in simulation. Notice that the basic OCM is immediately distinguished as the lower portion of the dashed block labelled pilot model. The upper portion of the pilot model displays the form of the expanded perceptual model. Observe that output signals from the simulator pass through dynamical blocks representing the visual and vestibular sensory systems of the human (such as inner ear dynamics) to form two display vectors, one from each modality. The displayed signals are then combined via a monitor. The monitor allocates attention to the individual elements, to form the usual display vector. The other blocks in Figure 1 represent the simulator hardware in a straightforward and conventional fashion. The main frame digital computer is assumed to generate the vehicle dynamics and its characteristics are a part of that block. Likewise the display computer characteristics are included in the simulator drive logic block. In this study, stick dynamics and stick transducer dynamics (such as computer generated force loading) were not considered. Also, visual flight was assumed so flight instruments could be ignored.

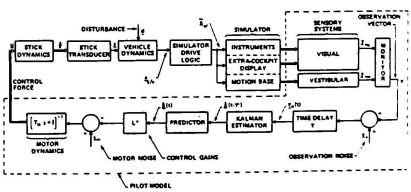


Figure 1: Overall Pilot/Vehicle System

Below, each aspect of the model definition is discussed. First, a brief description of the vehicle and control task is given. This is followed by a statement of the basic simulator (main frame computer, VMS, and CGI) characteristics and how they were modelled. A simple model for the perception of the visual scene will then be discussed along with the vestibular models. Finally, a summary of model parameters and assumptions will be presented.

Task/Vehicle Description

The pilot's task is to hover over a fixed point at a fixed altitude, in the presence of disturbances generated by air turbulence. Control is to be maintained by relying on extra-cockpit visual cues obtained from an out-the-window view and by motion cues associated with helicopter rotation and translation. Visual cueing is provided by a computer generated image (CGI) system, and motion cueing is provided by a vertical motion simulator (VMS).*

The specific helicopter chosen for this study was the CH-47 tandem rotor transport helicopter. The linearized and decoupled equations of motion for the helicopter, as well as the Drygen gust models for the air turbulence, were obtained from Hoffman et al.[5] The reader is referred to this report for specific details concerning the basic airframe equations.

The hovering task is modelled as a disturbance regulation task. As is standard procedure for application of the OCM, it is assumed that the objective of the task may be characterized as minimization of the following cost functional[3]:

$$J = E\left[\left(y_i/y_{i_{max}}\right)^2 + r_j u_j^2\right] \qquad (1)$$

where $y_{i_{max}}$ is a performance tolerance on the corresponding variable. The values for $y_{i_{max}}$ were chosen to be 5 ft and 1 ft/sec for position (x,y,z) and velocity (ẋ,ẏ,ż) variables and 1 deg and .05 deg/sec for attitude (ψ,θ,ϕ) and attitude rate ($\dot{\psi},\dot{\theta},\dot{\phi}$) variables; these values were taken from reference 5. The weightings on control rate activity, r_j, were chosen by means of an error-control tradeoff analysis. This resulted in a value of approximately .1 for the diagonal elements of the T_N matrix.

It should be noted that hover control of the unaugmented CH-47 is not an easy task. The results of the reference cited above suggest that the task cannot be performed to within acceptable tolerances under IFR conditions.

Main-Frame Computer

The vehicle equations of motion were implemented on a digital computer, operating at a nominal update rate of 30 Hz. Based on results from the analytic study by Baron et al,[6] we assumed for simplicity that the integration routine introduced no "distortion" in the continuous vehicle dynamics being modelled, and that the only effect of digitization was the introduction of a sample and hold delay associated with the base cycle time of the main-frame computer. (See Figure 2)

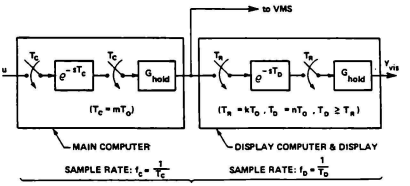


Figure 2: Visual and Motion Path Delays

CGI Characteristics

Table 1 summarizes the characteristics used to define the nominal CGI configuration. The nominal field-of-view specification is illustrated in Figure 3 as screen configuration B. Notice from the table that no dynamics were associated with the CGI system.

Table 1: Nominal CGI Characteristics

Picture Refresh Rate	30 frames /s
Display Compute Time	66 msec
Effective Sample Rate/Delay	15 Hz/99 msec
Scene Content	6000 edges /frame
Field-of-View	3 screens across (144° horiz, 36° vert)
Display Resolution	104 lines/frame x 1024 pixels/line

The graphics computer was modelled in the same fashion as the main frame computer; as a sample and hold delay based on the refresh rate and a compute delay (Figure 2). Because the display computer is in series with the main frame computer, the total visual delay must include the main frame delay. The effective sample rate is simply the slowest component in this path (i.e., the fastest rate at which information can change). This determines the effective visual hold time for information so that the total visual delay can simply be written as:

$$T_{vis} = T_c + T_d + 1/(2f_{eff}) \qquad (2)$$

where T_c is the base cycle time of the main frame computer, T_d is the display compute time, and f_{eff} is the effective sample rate. $T_{vis} - 3T_c$ seconds of the total delay was modelled as a Pade' delay in the visual path, while the rest of the delay was lumped into the human's time delay. (Since the main frame delay, $\frac{3T_c}{2}$ is common to both the VMS and the CGI).

*In spite of its name, the VMS is not restricted to vertical motion cues; it is a six degree-of-freedom cueing system.

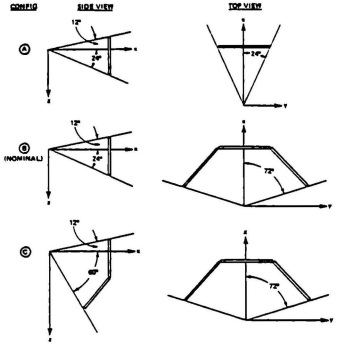


Figure 3: CGI Screen Configuration

Scene content was not modelled except insofar as it was needed in the assumptions for the visual perception model. Field of view serves to limit the utility of the displayed information and its effect will be seen when we discuss the pilot's perception of the visual scene. Screen resolution limits the fineness of detail the CGI system is able to present. Therefore, we have modelled it as a threshold equal in value to the average vertical and horizontal angular resolution level. This threshold is, of course, in addition to any human operator thresholds, and will be described shortly.

VMS Characteristics

The nominal VMS was modelled in all axes as a second-order dynamic model with appropriate position/rate/acceleration servo limits. Table 2 defines the nominal parameter values associated with each motion axis. The nominal motion system did not include washout filters as all predicted motions except the surge (x) motion were well within their respective simulator limits. In addition to dynamical models for the VMS servo system, there is an effective motion system delay due to the main frame computer.

Table 2: VMS Model Parameters

LINEARIZED TRANSFER FUNCTION: $\frac{\lambda}{\lambda_c} = \frac{\omega_n^2}{s^2 + 2\zeta_n\omega_n s + \omega_n^2}$

PARAMETERS:

AXIS	ω_n*	ζ_n	$\ddot{\lambda}^{max}$	$\dot{\lambda}^{max}$	λ^{min}	λ^{max}
ROLL (ϕ)	9.4	0.7	$50^0/s^2$	$15^0/s$	-22^0	22^0
PITCH (θ)	9.4	0.7	$50^0/s^2$	$15^0/s$	-24^0	26^0
YAW (ψ)	9.4	0.7	$50^0/s^2$	$15^0/s$	-29^0	29^0
SURGE (x)	9.4	0.7	$16FT/s^2$	2FT/s	-2.5FT	2.5FT
SWAY (y)	18.8	0.7	$24FT/s^2$	10FT/s	-20FT	20FT
HEAVE (z)	18.8	0.7	$32FT/s^2$	20FT/s	-30FT	30FT

* RAD/S

Visual Perception Model

In contrast to the relatively well-defined set of visual cues provided by within-cockpit instrumentation, the extra-cockpit visual scene can provide the pilot with an exceptionally rich stimulus environment, even for a relatively simple display. Attempting to describe and quantify this stimulus environment in some detail has been the object of many studies[7,8] and is well beyond the scope of this paper.

One approach to modeling the visual scene is to use the perspective geometric arguments of Wewerinke.[9] Then, each object of importance is modelled as a series of line segments, and is incorporated into the display matrix C. As a result, each object can yield many cues. In Wewerinke's study of the approach and landing of a conventional aircraft there were a limited number of well-defined line elements comprising the visual scene, and thus the construction of the display matrix C was a relatively straightforward exercise. However, in the hover task, no single object is as important as the runway in the approach and landing situation. Instead, a pilot can use various portions of the visual field, and any number of objects or parts of objects to maintain hover position and attitude. Furthermore, if it is assumed that a relatively "realistic" visual scene is available to the pilot, such a scene would typically be comprised of thousands (or perhaps tens of thousands) of discriminable line elements (and hence cues). This would make the display analysis used by Wewerinke intractable and, consequently, his approach was not used here.

Our approach, instead, was to take a much simplified view of visual cue processing, based on the following notion. Information from cues involve changes in the location, and/or orientation of the various line elements comprising the visual scene. These changes, in turn, can be expressed in terms of changes to the angular coordinates associated with the line element, two coordinates per point. For our study, we have taken these two coordinates to be the azimuth and elevation angles associated with the line-of-sight (LOS) to a particular line element endpoint. This is illustrated in figures 4 and 5, which show how specific vehicle rotations and translations result in changes in the azimuth and elevation angles associated with the line-of-sight to specific points in the visual scene.

Changes in the LOS angles are due to changes in vehicle state (position and attitude). Assuming small changes, we can use linearized relations, so that

$$\psi_{vis} = c_\psi \underline{x}' \quad (3a)$$

$$\theta_{vis} = c_\theta \underline{x}' \quad (3b)$$

where ψ_{vis} and θ_{vis} are the azimuth and elevation LOS angles, c_ψ and c_θ are the display "gains", and it is understood that the above relation holds (with different gains) for each specific point in the visual scene. We then assumed that for each vehicle state the pilot was trying to estimate, that he would chose one particular

point in the visual scene to provide the most appropriate visual cue, and then share his attention among these now competing cues. Thus, if the vehicle state is comprised of three rotational coordinates and three translational coordinates, then, in general, there would be six specific points in the visual scene the pilot would use for inferring changes in vehicle states. Notice there is a many on one mapping between the vehicle states and each point in the scene. However, rather than postulating a cue decoupling model, it was assumed the pilot was able to perform the inverse transformation needed to infer correct vehicle state information from the many observations available.

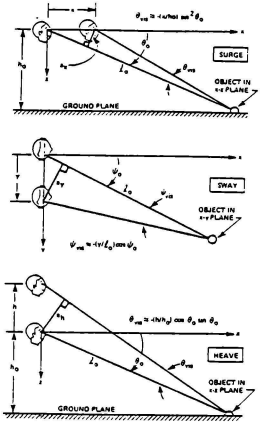


Figure 4: LOS Changes Due to Translation

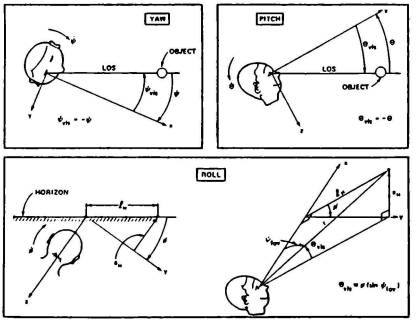


Figure 5: LOS Changes Due to Rotation

It is now appropriate to consider the fact that the pilot will be limited in his ability to detect changes in the LOS angle cues available to him. This limitation will be due either to his own inherent sensory/perceptual limitations, or, in the simulator situation, to CGI-imposed resolution limits. The effective visual cue threshold will be the greater of the two thresholds (sensory or display hardware), and will ultimately limit the pilot's ability to infer vehicular state changes from changes in the visual scene. Naturally, if display hardware is not involved (as in the actual helicopter environment), then the effective threshold will be determined solely by the pilot's visual limitations.

Turning first to the pilot's visual limitations, we make a distinction between angular resolution threshold (α_R) and angular discrimination threshold (α_D). The former refers to visual acuity, and the ability to resolve small angular differences in the LOS angle, when given a visual reference which, in angular distance, is very close to the object being sighted. The latter refers to the ability to discriminate between two large visual angles, and thus to identifying a small angular difference in the LOS angle, when given a visual reference which, in angular distance, is relatively far from the object being sighted.

The angular resolution threshold (α_R) might be chosen on the basis of measured human visual acuity, which appears to be on the order of one minute of arc.[10] However, we chose to set it at a slightly higher level, based on an earlier analysis of the data obtained from dynamic tracking experiments:[11]

$$\alpha_R = 0.05 \text{ deg}$$

The angular discrimination threshold (α_D) was chosen in accordance with the Weber-Flechner law[12] and set at a fixed fraction of the total angle being viewed:

$$\alpha_D = {}_o/30$$

where α_o is the total angle being viewed.

We now define the pilot-associated visual threshold as the maximum of the resolution and discrimination thresholds:

$$\alpha = \text{MAX}(\alpha_R, \alpha_D) \qquad (4)$$

The overall pilot/simulator visual threshold will be given by

$$\gamma = \max(\alpha, \beta) \qquad (5)$$

where α is the effective pilot threshold obtained previously, and the overall simulator threshold is:

$\beta = (\beta_H + \beta_V)/2$
β_H = horizontal CGI resolution threshold
β_V = vertical CGI resolution threshold.

The discussion to this point has concentrated on static "position" thresholds. To determine dynamic "velocity" thresholds associated with visual cueing, one might attempt

to assign a value on the basis of past psychophysical motion detection/discrimination experiments. However, a review of the subject by Graham[13] shows that a wide range of values can be assigned, depending on the particular experimental situation and empirical measures used. We again chose to assign a value on the basis of the earlier dynamic tracking experiments.[11] These studies yielded a ratio of approximately 4:1 between velocity and position thresholds. Thus, for this study we chose the visual velocity threshold according to:

$$\dot{Y} = 4Y$$

where, if Y is given in degrees, $\dot{Y}$ is given in degrees/sec. Hence, we tie the rate threshold to the position threshold, which, in turn, depends on the pilot-associated and display-associated resolution limitations. These ideas are summarized below.

We are now in a position to define the effective "informational" thresholds, associated with the visual cues available to the pilot. Recall, we assume the pilot can "invert" the appropriate display equations to obtain an estimate of the vehicular attitude/position change from the visual cues available to him. If we assume that the effective visual threshold applies equally to the azimuth (ψ_{vis}) and elevation (θ_{vis}) LOS changes, we can use figures 4 and 5 to generate informational threshold functions as shown in Table 3.

Table 3: Visual Scene Informational Function

AXIS	POSITION	VELOCITY
YAW	$\psi_{TH} = \gamma$	$\dot{\psi}_{TH} = \dot{\gamma}$
PITCH	$\theta_{TH} = \gamma$	$\dot{\theta}_{TH} = \dot{\gamma}$
ROLL(1)	$\phi_{TH} = \left(\frac{1}{2\sin\psi_{FOV}}\right)\gamma$	$\dot{\phi}_{TH} = \left(\frac{1}{2\sin\psi_{FOV}}\right)\dot{\gamma}$
SURGE(2)	$x_{TH} = \left(\frac{h_o}{\sin^2\theta_o}\right)\left(\frac{\theta_o}{30}\right)$	$\dot{x}_{TH} = \left(\frac{h_o}{\sin^2\theta_o}\right)\dot{\gamma}$
SWAY(3)	$y_{TH} = l_o\gamma$	$\dot{y}_{TH} = l_o\dot{\gamma}$
HEAVE(2)	$z_{TH} = \left(\frac{h_o}{\sin\theta_o\cos\theta_o}\right)\left(\frac{\theta_o}{30}\right)$	$\dot{z}_{TH} = \left(\frac{h_o}{\sin\theta_o\cos\theta_o}\right)\dot{\gamma}$

(1) FACTOR OF 2 ACCOUNTS FOR TWO END-POINT HORIZON

(2) POSITION THRESHOLD IS DISCRIMINATION LIMITED

(3) ASSUMES VISUAL TARGET STRAIGHT AHEAD ($\psi_o = 0$)

This table relates visual scene thresholds to (displayed) vehicle state thresholds. To determine the specific values for these informational thresholds, first assume a nominal hover altitude (h_o) of 10 ft, and a nearest eye level visual target at a distance (l_o) of 50 ft. Note that the maximum lateral field of view (ψ_{FOV}) and LOS depression angle (θ_o) are both set by the screen configuration. Then, minimizing the threshold functions of Table 3 and solving for θ and ψ, for each screen configuration and display resolution considered, determines the "best" viewing locations along with the values for the informational thresholds. The resulting thresholds are summarized in Table 4.

Table 4: Visual Scene Informational Threshold Values

THRESHOLD		RESOLUTION: 500 LINES/FRAME			1200/2000 LINES/FRAME		
		CONFIG.	A	B	C	A	B**	C
ROTATION								
LONG.	Θ_{TH}		.38	.08	.08	.05	.05	.05
	$\dot{\Theta}_{TH}$		.34	.34	.34	.20	.20	.20
LAT.	Ψ_{TH}		.08	.08	.08	.05	.05	.05
	$\dot{\Psi}_{TH}$		.34	.34	.34	.20	.20	.20
	Φ_{TH}		.10	.04	.04	.06	.02	.02
	$\dot{\Phi}_{TH}$		.42	.16	.16	.25	.09	.09
TRANSLATION								
LONG.	$\dot{x}_{TH}$		.54	.84	.47	.84	.84	.47
	$\ddot{x}_{TH}$		.36	.36	.08	.21	.21	.05
	$\dot{z}_{TH}$		.37	.37	.37	.37	.37	.37
	$\ddot{z}_{TH}$		.17	.17	.17	.10	.10	.10
LAT.	$\dot{y}_{TH}$		.57	.37	.37	.34	.04	.04
	$\ddot{y}_{TH}$		.30	.30	.30	.17	.17	.17

* ROTATION THRESHOLD VALUES IN DEG & DEG/SEC
TRANSLATION THRESHOLD VALUES IN FT & FT/SEC

** NOMINAL CONFIGURATION

In summary, since the perceptual dynamics of the human visual system are relatively wide-band with respect to the system dynamics we are modelling, we chose not to include any dynamic visual effects. This allowed us to implement our visual perception model by simply specifying thresholds for the appropriate system state variables: the linear/angular positions and velocities of the (simulated) vehicle.

Vestibular Perception Model

Models of vestibular motion perception have been the subject of study for a number of years, and we will not attempt to summarize this work. Instead, we refer the reader to a relatively recent review of motion cue models by Zacharias,[14] in which a number of these models are described and critically reviewed.

Figure 6 shows the vestibular model in block diagram form. The upper portion models the semi-circular canals as transducers of angular velocity, while the lower portion models the otoliths as transducers of specific force. Table 5 sumarizes the parameter values used in each of the vestibular models.

To reduce computational requirements imposed by the vestibular model, we performed an analysis of the power spectrum of the vestibular signals. By comparing the power spectra of incoming vestibular signals to that of their filtered outputs, pass-bands were identified which accounted for the majority of the correlated power. Utilizing this information allowed the elimination of any lead or lag elements having break frequencies not in the pass-bands. Table 6 outlines the resulting simplifications. Although many of the vestibular dynamics were simplified or eliminated, the vestibular thresholds given in Table 5 were still implemented.

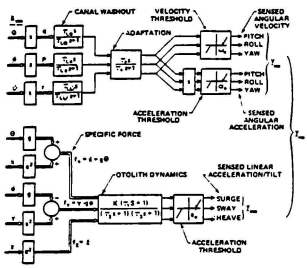


Figure 6: Vestibular Model

Table 5: Parameter Values for Vestibular Model

Canal Parameters

AXIS	τ_L(DEG/S)	ω_0(DEG/S)	α_0(DEG/S^2)	
PITCH (θ)	5.3	3.6	0.67	τ_a = 30s
ROLL (ϕ)	6.1	2.5	0.41	
YAW (ψ)	10.2	4.2	0.41	

Otolith Parameters

τ_1 = 13.2s, τ_2 = 5.33s, τ_3 = 0.66s
K = 0.4, a_0 = 0.053FT/S^2 } ALL AXES

Table 6: Simplifications to the Vestibular Model

Axis	Simplification
Pitch (θ)	Eliminate canal washout and adaptation filters
Roll (ϕ)	Eliminate adaptation filter
Yaw (ψ)	Eliminate canal washout and adaptation filters
Surge (f_x)	No simplification
Heave (f_z)	Set $\tau_1 = \tau_2 = 0$
Sway (f_z)	No simplification

Attention-Sharing Model

The general features and method of implementation of the attention-sharing model are well known.[15,16] Here, we wish to describe features of the model which are specific to the particular helicopter hover task under consideration.

In our modelling of the hover task, we assumed that "full attention" corresponds to an overall noise/signal ratio of -20 dB, a level which is consistent with the finding of many earlier manual control studies.[3] Further, it was assumed there would be an optimum allocation of attention among the displayed variables subject to several constraints. These constraints involve assumptions about attention-sharing between tasks and between perceptual modalities. Thus, although the lateral axis control task is more demanding than the longitudinal axis,[5] for the purpose of this study we assumed an equal split of attention between the two axes. In addition, it was assumed that it was not necessary to share attention between modalities (i.e., visual and vestibular signals are processed in parallel). This requires that within an axis the total visual attention equal the total vestibular attention. This last assumption will clearly favor the use of motion cues, provided they are useful for control, since they provide added information essentially without cost.

The various aspects of the pilot/vehicle/simulator model are summarized in Table 7.

Table 7: Summary of Model Implementation Characteristics

- **TASK OBJECTIVE**
  MAINTAIN FOLLOWING RMS HOVER ERRORS:
  ATTITUDE } 1 DEG
  ATTITUDE RATE } 0.5 DEG/S
  POSITION } 5 FT
  VELOCITY } 1 FT/S
- **INFORMATION-PROCESSING/CONTROL-BANDWIDTH LIMITATIONS**
  OBSERVATION NOISE/SIGNAL RATIO -20 dB
  INTERNAL TIME DELAY 0.20 S
  MOTOR TIME CONSTANT 0.15 S
- **VISUAL PERCEPTION MODEL**
  PERSPECTIVE/GEOMETRIC CUES
  NO SENSORY DYNAMICS
  RESOLUTION/DISCRIMINATION THRESHOLDS
- **MOTION PERCEPTION MODEL**
  ROTATIONAL AND SPECIFIC FORCE CUES
  VESTIBULAR DYNAMICS (CANALS & OTOLITHS)
  RESOLUTION THRESHOLDS
- **ATTENTION-SHARING MODEL**
  SHARED ATTENTION BETWEEN LONGITUDINAL AND LATERAL AXES
  NO INTERFERENCE BETWEEN MODALITIES
  OPTIMUM SHARING WITHIN MODALITIES

Closed-Loop Analysis of CGI and VMS Effects

In this section, the optimal control model with the expanded perceptual model is used to analyze the effects of CGI and VMS limitations on closed-loop hover performance. The goal of this analysis is to determine the effects of CGI and VMS characteristics on simulator fidelity (more precisely, performance and workload). To this end, a "perfect" or ideal simulator is defined in which there are no simulation time delays, no motion system dynamics, and an infinite resolution imagery system. This simulator configuration corresponds essentially to flight* and provides a benchmark against which to measure simulator deficiencies. In addition to the nominal and perfect motion conditions, results were also obtained for a "no-motion" or fixed-base simulator configuration.

Thus, there were six basic simulator configurations to be analyzed so as to evaluate the effects of the visual and motion systems, separately and together. These configurations

*Through an oversight, the assumptions for the perfect configuration included a field-of-view constraint relevant to the nominal CGI configuration. This degraded performance only slightly from what would have been obtained without the constraint.

are listed in Table 8.

Table 8: Simulator Configurations

CONFIGURATION	DESCRIPTION
Perfect (Flight)	No simulator delays, nominal field of view, human operator thresholds, no VMS dynamics
Perfect CGI-Realistic VMS	Includes main frame computer delays and VMS platform dynamics
Realistic CGI-Perfect VMS	Includes main frame and display computer delays, CGI imposed visual thresholds, no platform dynamics
Realistic CGI-Realistic VMS	Includes all simulator nominal characteristics (see Table 5.1)
Perfect CGI-Fixed Base	Includes main frame computer delays in visual cues, no motion cues
Realistic CGI-Fixed Base	Includes CGI limitations, no motion cues

Results and Discussion

The effects of CGI and motion system characteristics will be examined largely in terms of relative performance in the hovering task. For each axis, relative performance is defined as

$$\text{Performance (in \%)} = 100 \times (J - J_{FLT})/J_{FLT} \quad (6)$$

where J is the value of the cost functional of Eq. 1 and J_{FLT} corresponds to the value of J obtained for flight or the "perfect" simulator. Thus, relative performance is a normalized metric of performance that measures the percent deviation from "flight" performance introduced by the simulator characteristics. In this sense, relative performance is a measure of simulator fidelity.

The results will be presented in terms of J (rather than individual error and control scores) because this quantity is a scalar metric of overall performance and, therefore, provides a concise description of the simulator effects. In addition, Hess[17] has shown that the value of J may be correlated with vehicle flying qualities, so increases in J owing to simulator deficiencies may be related to degraded flying qualities for the simulator.

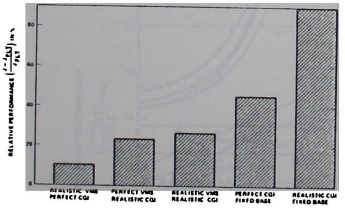
Overall CGI and Motion System Effects

Figure 7 presents performance predictions of the model for the five simulator configurations, relative to that expected from the "perfect" simulator (which, by definition, has a relative performance of zero). With respect to longitudinal performance, it can be seen that the effect of the CGI is much more significant (35%) than that of the motion system (10%). Indeed, performance is better with a perfect CGI and no motion than with perfect motion and a realistic CGI. However, motion is still important, particularly if the realistic CGI deficiencies are accounted for. This is shown by the prediction of approximately twice the relative performance for the realistic CGI-fixed base configuration as for the realistic CGI-realistic VMS configuration.

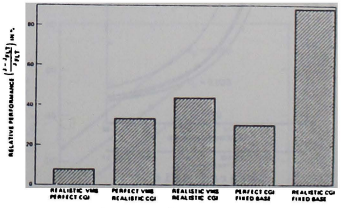
The results for the lateral control task are similar to those for the longitudinal task, but motion is even more important. In this case, having a perfect CGI does not compensate for lack of motion, since the fixed base configurations are worse than any other motion configuration. Compared to the longitudinal task, going from perfect to realistic motion introduces less performance degradation. Also, motion ameliorates the consequences of any visual deficiencies.

For either longitudinal or lateral control, the performance change (10-15%) due to introducing the realistic motion system alone is probably within the inter- and intra-pilot variations that might be expected. However, once realistic CGI effects are considered, or motion is removed entirely, this is no longer likely to be true for skilled pilots inasmuch as the deviations predicted can be substantially greater than 20%.

The above model predictions are based on the assumption that the pilot will maintain a fixed level of attention for the longitudinal and lateral control tasks regardless of simulator configuration. However, in actuality, the pilot may choose to devote more (or less) attention to the control tasks, based on simulator configuration. To explore the effects of such a change in strategy, model predictions were obtained for various attention levels. The



Lateral Axis



Longitudinal Axis

Figure 7: Relative Performance vs. Simulator Configurations

results are presented in Figure 8. Note that the solid dots on the curves indicate the nominal level of attention for that simulator configuration. It can be seen that the relative ordering of simulator configurations is maintained at all levels of attention. At high levels of attention, the performance with the realistic CGI-perfect VMS configuration approaches that for the realistic VMS-perfect CGI configuration. Apparently, if the noise/signal ratio is lowered sufficiently on the motion cues, it can offset some of the visual deficiencies associated with the nominal CGI.

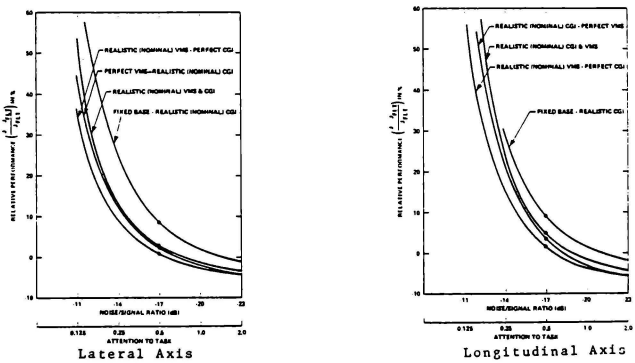
If it is assumed that the pilot adapts his behavior and increases attention levels to achieve performance equivalent to that in flight, then the incremental attention required may be considered a workload penalty associated with the simulator. The curves of Figure 8 can be used to determine this workload penalty for maintaining flight level performance in the simulator; one simply determines the intersection of the particular sensitivity curve with the line of zero relative performance. The computed attention or workload penalties for the various configurations analyzed in Figure 8 are given in Table 9. For the nominal CGI and motion system, the pilot would have to increase attention by 50% over that needed in flight in order to achieve the same performance, whereas almost three times as much attention is required for a fixed base simulation.

Effects of CGI Parameters

The results of the previous section suggest that the visual processing limitations introduced by a nominal CGI configuration could result in significant deteriorations of closed-loop hover performance. Here, we examine the effects of variations in individual, design-related CGI parameters. In these analyses, a single parameter is varied while all other CGI parameters are kept at their nominal or realistic values. Results will be presented for both realistic motion base and fixed base configurations.

Figures 9 and 10 show the effect of incremental delays on relative performance for motion-base and fixed-base simulators, respectively. Results are presented as a function of CGI display computer delay, for three values of main-frame computer delay (T_c). Recall, the nominal display delay is 99 msec. For the range of delays considered, relative performance appears to degrade linearly as a function of either display delay or main-frame delay, when motion is present. Comparison of Figures 9 and 10 (note the difference in scale) reveals that the absence of motion cues will accentuate the deterioration of performance for a given delay. Moreover, for a fixed base configuration, performance degrades more rapidly than linearly. It can also be seen from these figures that the longitudinal control task is more sensitive to increases in delay than is the lateral task, particularly to increases in display delays.

In general, the magnitude of the effects of display delay are quite significant. Increasing display delay from zero to the nominal, but reasonably conservative, value of 99 msec, causes an increase in relative performance of approximately 20-30% for the motion-base simulation and about 40-50% for the fixed-base case. An examination of the relative performance values for zero display and computer delay shows that the effects of other CGI or motion system limitations are much less significant (at nominal values) than are the effects due to delays.



Lateral Axis Longitudinal Axis

Figure 8: Relative Performance Vs. Workload

*The possible effects of increased field-of-view providing useful peripheral information on vehicle rates have not been examined here.

Table 9: Simulator Workload Penalties

CONDITION	- ATTENTION -		
	LONG.	LAT	TOTAL
	.5	.5	
R. VMS, P. CGI	.55	.58	1.0
P. VMS, R. CGI	.66	.69	1.35
R. VMS, R. CGI	.76	.76	1.52
R. CGI - F. B.	1.25	1.5	2.75

The effects of field-of-view and display resolution are presented in Figure 11. Recall that screen configuration B is the nominal configuration corresponding to a 144° horizontal, 36° vertical field of view. Configurations A and C provide 48° by 36° and 144° by 72° fields of view, respectively. The nominal display resolution is 1024 lines. Both field of view and display resolution are assumed to affect observational thresholds as discussed previously.

The effects of display resolution are somewhat different than for field-of-view in that a greater effect is observed for the longitudinal task than the lateral task. With motion, longitudinal performance is about 20% poorer for the 500 line display as compared to about a 5% degradation in the lateral case; for the fixed-base configurations, these effects are increased to about 25-30% and 10%, respectively.

It can be seen from Figure 11 that decreasing the horizontal field of view (configuration A) does not affect longitudinal performance and increasing the vertical field-of-view has no effect on lateral performance. This is expected because of the assumed decoupling between longitudinal and lateral control tasks. Figure 11 also suggests that increasing vertical field-of-view has very little performance payoff and probably would not be justified on the basis of these results. On the other hand, the improvement in performance with increased lateral field-of-view appears to be significant, especially if the cue presentation is degraded in other ways, such as poorer resolution or no motion. For the 500 line display, fixed base configuration, reduction of the horizontal field-of-view from 144° to 48° degrades relative performance by more than 30%.

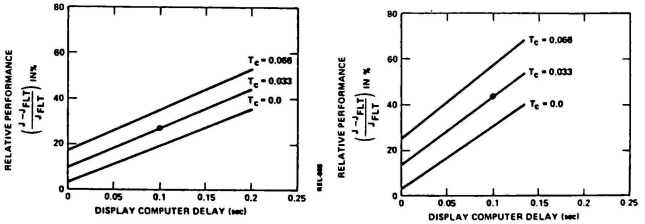
Before leaving this discussion of the effects of individual CGI parameters, it should be noted, as a caution, that the assumption of a one-to-one correspondence with model parameters is made for simplicity. In reality, design changes can alter several factors related to information processing and tradeoffs are often the result. For example, improved scene content may lower noise/signal ratios but may require more computation and, hence, increase delay.

Effects of VMS Parameters

Relative performance is plotted as a function of platform bandwidth and control task in Figure 12. A bandwidth of zero corresponds to a fixed base configuration and an infinite bandwidth corresponds to flight motion. It can be seen that changing the bandwidth does not have an appreciable effect on relative performance, so long as a reasonable degree of motion fidelity is maintained. The effects of bandwidth are somewhat more pronounced for the longitudinal control task than for the lateral.

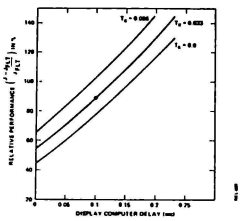
Effects of Vehicle Dynamics

The effects of simulator parameters will depend on the specifics of the task, including the vehicle dynamics. This has already been illustrated in differences between predicted longitudinal and lateral performance. To explore further the effects of vehicle dynamics, results were obtained for the CH-47 with a velocity command control augmentation system, as specified in Hoffman et al[5] as system F. The augmented vehicle presents a significantly less difficult control task. Figure 13 gives relative performance as a function of control augmentation for the nominal simulator configuration (and for the nominal fixed-base configuration). The effect of simulator characteristics is substantially less for the augmented vehicle. However, the effect is still significant for longitudinal control and for fixed-base simulation of lateral (augmented) control.

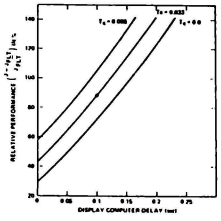


Lateral Axis Longitudinal Axis

Figure 9: Relative Performance vs. Time Delay (moving base simulator)

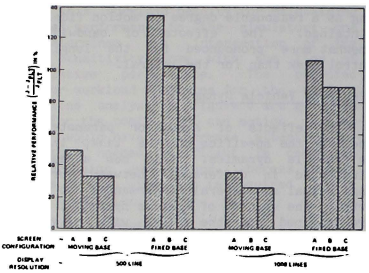


Lateral Axis

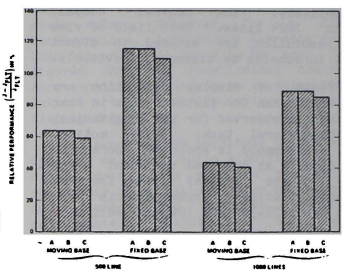


Longitudinal Axis

Figure 10: Relative Performance vs. Time Delay (fixed base simulator)



Lateral Axis



Longitudinal Axis

Figure 11: Relative Performance vs. Field of View and Display Resolution

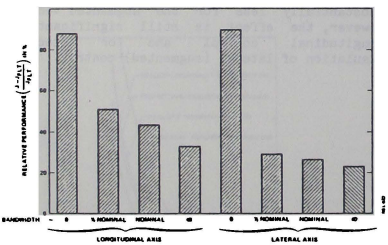


Figure 12: Relative Performance vs. VMS Platform Bandwidth

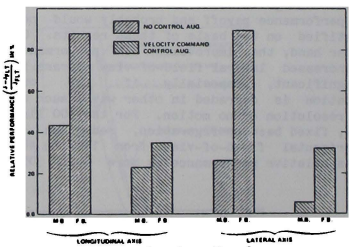


Figure 13: Relative Performance vs. Control Augmentation

Summary And Conclusions

The optimal control model for pilot/vehicle analysis has been used to explore the effects of a CGI visual system and motion system dynamics on helicopter hover simulation fidelity. This was accomplished by expanding the perceptual aspects of the model to include motion sensing and by relating CGI parameters to information processing parameters of the model. Simulator fidelity was examined by comparing predicted performance and workload for flight with that predicted for various simulator configurations.

The results of the analysis suggest that simulator deficiencies of a reasonable nature (by current standards) can result in substantial performance and/or workload infidelity. Both CGI and motion system effects are significant for this task. There is an interaction between the two sources of pilot cues. In particular, the presence of motion reduces the sensitivity to CGI limitations.

With respect to the CGI system, the most important parameter in terms of its effect on performance was display delay. This was followed in order of importance by display resolution and field-of-view.

The main effect associated with motion system bandwidth was introduced by going to a fixed-base configuration. Halving the VMS platform bandwidth or going to full flight motion made only a marginal change in the performance predicted for the nominal VMS bandwidths.

The trends of the results are fairly consistent although there were some differences between lateral and longitudinal control tasks. The magnitude of the effects and relative importance of various parameters are clearly dependent on the task as exemplified here by longitudinal vs. lateral and unaugmented vs. augmented vehicle dynamics. It is, of course, for this reason that models of the pilot/vehicle system are needed to evaluate the importance of simulator parameters for a given situation.

References

1. Gum, D. R. and Albery, W. B., "Time Delay Problems Encountered in Integrating the Advanced Simulator for Undergraduate Pilot Training", J. of Aircraft, 4, 1977.

2. Quiejo, M. J. and Riley, D. R., "Fixed-Base Simulator Study of the Effect of Time Delays in Visual Cues on Pilot Tracking Performance", NASA TN D-8001, October 1975.

3. Kleinman, D. L., Baron, S. and Levison, W. H., "A Control Theoretic Approach to Manned-Vehicle Systems Analysis", IEEE Trans. on Auto. Control, AC-16, 1971.

4. Baron, S., "A Model for Human Control and Monitoring Based on Modern Control Theory", Journal of Cybernetics and Information Sciences, Vol. 1, No. 1, Spring 1976, pp. 3-18.

5. Hoffman, W. C., Kleinman, D., and Young, L., "Display/Control Requirements for Automated VTOL Aircraft", ASI-TR-76-39, October 1976.

6. Baron, S., Muralidharan R. and Kleinman, D. L., "Closed Loop Models for Analyzing the Effects of Simulator Characteristics", AIAA Flight Simulation Tech. Conf., Arlington, TX, September 1978.

7. Gibson, J. J., The Perception of the Visual World, Riverside Press, Cambridge, Mass., 1950.

8. Brown, J. L., "Visual Elements in Flight Simulation", University of Rochester, TR-73-2, December 1973.

9. Wewerinke, P. H., "A Theoretical and Experimental Analysis of the Outside World Perception Process", Proceedings of the Fourteenth Annual Conference on Manual Control, NASA conf. Pub. 2060, November 1978.

10. Riggs, L. A., "Visual Acuity", in Vision and Visual Perception, C. H. Graham (editor), John Wiley and Sons, NY, 1965.

11. Levison, W. H., "The Effects of Display Gain and Signal Bandwidth on Human Controller Remnant", AMRL-TR-70-93, Wright-Patterson Air Force Base, Ohio, March 1971.

12. Luce, R.D. and Galanter, E., "Discrimination", in Handbook of Mathematical Psychology, R.D. Luce, R.R. Bush, and E. Galanter (editors), Vol. 1, John Wiley and Sons, Inc., New York, 1963.

13. Graham, C. H., "Perception of Movement", in Vision and Visual Perception, C. H. Graham (editor), John Wiley and Sons, N.Y., 1965.

14. Zacharias, G. L., "Motion Cue Models for Pilot-Vehicle Analysis", AMRL-TR-78-2, Wright-Patterson Air Force Base, Ohio, May 1978.

15. Levison, W. H., Elkind, J. I., and Ward, J. L., "Studies of Multivariable Manual Control Systems: A Model for Task Interference", NASA-Ames Research Center, NASA CR-1746, May 1971.

16. Kleinman, D. L., "Solving the Optimal Attention Allocation Problem in Manual Control", IEEE Trans. on Automatic Control, AC-21, 1976.

17. Hess, R., "Prediction of Pilot Opinion Ratings Using an Optimal Pilot Model", Human Factors, 19, 1977.

81-0973

MICROCOMPUTER BASED ENGINE MODEL USED IN FLIGHT SIMULATION APPLICATIONS

François E. Huguenin*
Reishauer Ltd.
Electronics Departement
CH-8304 Wallisellen, Switzerland

Abstract

The objectives of this paper is to present and discuss some interesting aspects of modelling an aircraft engine using the latest microcomputer technology. One of the design approach taken during the course of this developement work leads to a decentralized highly modular design concept which can be of great advantage in applications dealing with real-time flight simulation. The paper will be divided into five major parts.

First some aspects of microcomputer hardware design including multi-processor architecture will be presented, especially in the field of real-time flight simulation.

The second part deals with the problem of modelling a CF-6/50 aircraft engine using a microprocessor. The design of the engine model and the problems encountered, during realization, like real-time implementation or modular software design, will be discussed in detail.

The third part will briefly overview the software created for the engine model and the approach chosen for programming it.

The fourth part will present some results of the simulation. Special emphasis will be given to the comparision between the microcomputer based engine model and the real data provided by the AIDS (Aircraft Integrated Data System) of a DC-10/30 airplane. The interface to a DC-10/30 flight simulator and some results of the connection of it to the microcomputer will also be presented.

Finally some important conclusions related to experience of the use of microcomputers for flight simulation applications will be presented. Results of this microcomputer based engine model have shown that with the now available high performance 16-bit microcomputers, it is absolutely possible to design complex simulation programs and obtain very satisfactory results. This of course opens new interesting possibilities in the aera of flight simulation applications.

I Introduction

This paper presents the results of the design of a microprocessor based engine model for real-time simulation. The objectives of this design are to:

1) Realize an on-line simulation of all key parameters of an aircraft jet engine. The model is dependant of the conditions of use given by the pilot of the aicraft (take-off, reverse, transient between two steady state levels of power setting etc.). The status of variable load factors like hydraulic systems, antiicing etc., also influence directly the engine model and must be taken into account for the simulation.

2) Design an engine model on a microcomputer system.

3) Make comparisions between the model and the real data provided by the AIDS system, in this case especially for the fuel flow.

4) Obtain some important data about the use of microcomputers in this kind of applications (cycle time, program length and complexity).

The engine model was designed and implemented on a single board computer using a 16-bit 8086 microprocessor from INTEL [1]. A special interface was developed for connecting a model of a DC-10/30 cockpit instrument panel for displaying engine data. Included are also some switches which allow to set or reset the various loads of the engine (anti--ice, air-conditionning etc...). The development of this engine model was part of a research project at the Swiss Federal Institute of Technology which was started 1977 [2].

Despite of the fact that the entire development work had to be discontinued due to lack of support, effort is currently being made to connect the engine model to an CAE DC-10/30 flight simulator for testing the engine and display unit under realistic conditions. A two processor system will be used for this purpose.

II Aspects of microprocessor hardware design

The very fast development of the VLSI Technology during the last couple of years allow now the manufacturer to offer integrated computer boards known as Single Board Computers (SBC), which have all necessary hardware to stand-alone applications.

Normally the single board computer has the following hardware implemented: CPU, Memory, Interrupt controller and Input/Output. In addition to that some SBC's have the so-called dual port Memory controller for dual access on the Random Access Memory (RAM) (on and off-board). The block diagram of

* Member AIAA

such a computer board is presented on Fig.1. The Read Only Memory (ROM), the Input/Output peripherals (I/O) and the Programmable-Interrupt Controller (PIC) are tied to a local bus. An arbiter controls the access from the local or system bus to the RAM.

Important is to note that the CPU can be extended with a co-processor 1 for enhancement of the overall performance of the SBC. As co-processor now available for the 8086 CPU are the 8087 80-bit Floating Point Numeric Data Processor (NDP) and the 8089, a dual channel I/O processor. The local/system bus architecture is of a great advandage when using more than one SBC in parallel on the same system bus as shown in Fig.2.

Each of the processor card is a SBC extended with the dual port memory controller. Other boards may be connected to the system bus, like memory, I/O, process control interface. Each of these is in free access as common ressource to one of the two SBC computers. Priority is resolved in a Master/Slave principle, either serially as daisy chain or parallel with a priority resolution circuit. The great advantage of this architecture is the dual port memory concept which allows the two computers to communicate through the system bus together and share a common memory block. Each computer is running at full speed when working on his local bus only. In case of an access to the common resource, the CPU must gain control of the system bus through the arbiter, which is given by the implemented priority resolution network.

A special software instruction is provided in the CPU instruction set for programming the synchronization of the two (or more) computers accessing a common memory block [1] . The application software is burned on EPROMs (max. 32 Kbytes), allowing the computers to be used in stand-alone Original Equipment Manufacturer (OEM) applications.

The advantage of using this kind of modular hardware in flight simulation applications are twofold. First it is possible to develop self-contained building blocks which leads to a highly modular design. An example was presented in [2]. It is even possible to consider more complex structures of several computers sharing the common bus, each of them being dedicated to one specialized task. Such a system was partly proposed in 3 .

Second, this distribution of tasks among several processors can reduce the calculation time for a given task. This can be of great advantage in very time-critical applications. With the just introduced Numeric Data Processor NDP, each computer can be equipped with this VLSI. This offers of course a very big performance increase, both in calculation speed and mathematical precision.

In the following, a microprocessor based engine model will be presented. This model, despite its complexity, can be implemented on a SBC computer. This is a good example how the engine model, which is a part of every flight simulation program, can be integrated as a building block.

III The CF-6/50 Engine Model

The engine model is based on data tables provided by the engine manufacturer. Each of these tables describes the behaviour of the engine data in function of various parameters. The basic structure of the engine model is given by Fig.3. All important parts of it are defined by so-called systems. Four variables are input to the engine model:

- T the outside temperature (Ram air),
- m mach number,
- p the ambient pressure,
- TA the engine throttle angle.

TA is the engine control signal, T, m, p are parameters which are varying during flight. Five switches (on/off) are defined for taking account of the various engine loads like:

- Air conditioning,
- Horse power extraction,
- Anti-Ice,
- 8-14th stage bleed air switch.

These switches are normally set from the flight engineer desk in a standard position for the engine in use.

The engine model itself is defined by 8 system blocks:

- **Inlet Equations :** reduction of the three input parameters to standard day sea-level values.

- **Power Lever System :** computation of the demanded engine thrust (forward/reverse). This is a model of the throttle lever of the cockpit. The variable idle setting (on-ground or flight idle) is also computed.

- **N2 Transient :** simplified model of the fuel control unit of the engine and N2 (core speed).

- **N1 :** Model of the fan speed as function of N2 and mach.

- **Thrust :** Model of the thrust (forward and reverse) as function of N2, N1 and mach.

- **Fuel Flow :** Model of the fuel flow as function of N2, Altitude and mach. Total fuel used in the engine can be computed from the beginning of the simulation.

- EGT : Model of the exhaust gas temperature as function of N2 and mach.

- EPR : Model of the engine pressure ratio as function of N2 an mach.

Each of these system blocks contains additional data tables for computation of the corrective terms necessary to take account of the engine load given by the flight engineer switches. The whole model is basically designed for steady state operation between 80% N2 and 110% N2. Transients are approximed by the module **N2 Transient** and in the various system blocks. A digital filter with adaptive coefficients is used for the approximation of the turbine acceleration based on the data from the manufacturer. This approach could, in the meantime, be considered as good enough, compared to the flight data provided by the AIDS system of the aircraft.

However there is no provision made at the present time for starting the engine (cold start) nor to stop it. The model is automatically set upon initialization to ground idle. The limitation for the simulatiulon is only given by the ammount of fuel which is entered to the computer before starting the model.

A system block is shown in detail on Fig.4 and summarize how the engine model was designed. A main data table is interpolated to give the basic Fuel Flow (EWFA). A corrective term (delta EWFA1) takes into account the change of Fuel Flow of the aircraft is varying altitude. A second term (delta EWFA2) takes into account of the various loads connected to the engine (Anti-ice, Air-conditionning, etc...). The two corrective terms are the added to give the corected total Fuel Flow (EWFA). The refered total Fuel Flow is then corrected to the actual value. A totalizer allows to track the ammount of Fuel used during simulation time.

The whole engine model is constructed like the Fuel Flow Module. Basically, the structure is very simple, based on interpolation from look up tables.

Many problems were encountered during the design phase of this project. First data had to be digitalized by a HP-Plotter in order to obtain the data tables. Programs had to be developed for a PDP-11/60 minicomputer for this purpose. The obtained data were then transfered to the microcomputer development system by mean of a HP-Data-Casette.

An important problem to solve was the organization of the data tables in the computer memory and the interpolation routines necessary to compute the data. Three interpolation routines were created for this purpose. Because of the time constraint (total simulation for all three engines of the DC-10/30 in less than 50msec) special attention was given to the mathematical computation for avoiding unnecessay divisions or multiplications. Shift and add instructions were used instead of them. Despite of the fact that all programs were written at assembly level, no serious problems occured during design and test of the software. This was achieved thanks a highly modular approach for designing this appliaction software.

IV The software design problem

As mentioned before, the software for this application was designed on a highly structered way. Each system block was first considered as a self-contained module and designed like that. A test software for the HP-2648 Graphics Terminal was created in order to test the module. It was therefore possible to set some parameters, make a sweep of main input variables and plot the results to display. Thanks to this simple principle it was the possible to visualize all the data tables of the engine model for control purpose.

The basic flowchart of the Fuel Flow System Block is given on Fig.5. The program flow is very simple and straight forward.
The integration of the engine model was done in the same way. First all data for each System Block were saved in a object library. Each System Block program was then organized as a general purpose subroutine. The engine model program itself resulted in a sequential calling of 8 subroutines.

An interesting problem had to be solved for the simulation of all three engines of the DC-10/30 aircraft. A parameter block was created for each of all three engines. A fourth parameter block (working aera) was defined for the engine program itself. A simulation cycle looked like following:

-Copy parameter block of the engine number one to the working aera.

- Call engine model subroutine.

- Copy results back to engine number one parameter block.

- etc.

The programming of this system was very simple using index registers and block MOVE instructions. The structure of the three engine model is shown on Fig.6.

It should also be mentioned, that because of the fact that the tail engine of the DC-10/30 has other data than the wing engines during reverse, it was necessary to define two different data tables for reverse operation.

V Results

The three engine model was implemented on a single board computer SBC. A small test

program allowed to set a certain amount of fuel, set various parameters and run the simulation. The cycle-time achieved for all three engines was less than 30 msec, that is in the meantime 10 msec for each engine including the overhead for the parameter block transfer! This was one of the very interesting results of this application.

Based on data provided by the AIDS system of a DC-10/30 aircraft, it was then possible to compare the model with real data. A small computer program had to be created in order to refer the AIDS data to standard day/sea level conditions. A comparision is given in the following table for one AIDS data frame:

Data	AIDS refered	Model	%error
N1	109.5	102.3	-6.5%
N2	102.4	102.4	-----
EGT	861	870	+1%
FF	9087	9370	3%
EPR	6.32	6.62	0.5%

The percent error may provide from engine based differences from the typical data provided by the manufacturer, from some numerical round-off problems in calculations by the microcomputer or possibly from false setting of initial conditions of the model. In general the modelling of the engine can be considered as good.

A second interesting test was made, driving the model with a data stream identical to the one given by AIDS. The exact data for N2 was derived from AIDS and stored in a table. For each step (one second time frame) the model was "driven" by the N2 provided by AIDS. A special computer program made a plot of the output for N1, EGT, EPR and FF. The results of these tests are shown on Fig.7.

Based on this it is possible to summarize:

- N1 is exellent compared to the AIDS data.
- EGT is rising too fast compared to the installed engine of the aircraft.
- FF can be considered as a good model.
- EPR is a little bit too high in the steady state.

As conclusion it is possible to state that FF, EPR, and N1 can be considered as being directly function of N2 and that EGT only had to be corrected. On the other hand it must be considered, that this was only a first test of the model with the AIDS data and that a lot of work should be done to validate the model.

Next to a special interface was designed for connecting the engine model to a CAE DC-10/30 flight simulator. The computer configuration is shown in Fig.8.

A dual processor system is provided for this special application. Because the connection to the flight simulator is a time-critical problem, it is necessary to distribute the workload among two processors.

Computer number 1 is a master in this system and synchronizes every 50 msec with the flight simulator for data transfer. A special 16-bit I/O unit had to be designed for the connection to the flight simulator.

Computer number 2 is a slave in this system and is activated sequentially through a software flag. The I/O number 2 module is a special interface for driving a model of the instrument panel of the DC-10/30 cockpit. The level of all parameters (N1, N2, EPR, EGT and FF) are displayed using solid state electronics, in this case bargraphs. This instrument panel allows to follow what is going on in the microcomputer based engine model and make comparision with the original cockpit instruments inside the flight simulator.

The timing diagram for the two processor system is shown on Fig.9. The master computer inputs his data from the flight simulator. Next he makes some scaling on these data before starting through a software flag the engine model. As the next states of all three engines are calculated by CPU number 2, CPU number 1 computes all the data for driving the displays (unscaling, limitations, coding to BCD etc...). The display is then updated with one cycle delay. With a synchronization flag set by CPU number 2, the master is then able to transfer back data to the flight simulator. A complete simulation cycle is possible, thanks to the dual processor system, in less than 50msec.

VI Conclusions

This microcomputer based engine model demonstrates some of the possibilities offered by the new microcomputer technology. In this particular case of an engine model, it was possible to well structure the model a find a good way for implementing it on the computer system. The experience gained throughout creating this application clearly showed that:

- The use of microprocessors can be useful even for very complex tasks like on-line simulation.

- A highly modular system could be designed without too much overhead.

- Despite the critical time-constraints, it was possible to have a cycle time for one engine of less than 10 msec. This could be achieved thanks to optimization during the software design stage.

The dual computer system (now in development) will allow to test the engine model under very realistic conditions.

The engine model as a test case opens new interesting perspectives for using micro-

computers in flight simulators. The Single Board Computers (SBC) offer the great advantage of being standardized at hardware level. Their architecture allows to realize, with virtually no effort, multiprocessing systems avoiding all the problems encountered while designing dual port memory, bus controllers etc...

Some very specific modules of a flight simulator could be implemented on a microcomputer system. A lot of possibilites are open for: autopilots, flight instruments, engine models, navigation computers. Another field for potential applications could be the specific training parts of a flight simulator like instructor's console, on-line tracking of the simulator's state (altitude, specific flight data recording), or for repositioning the flight simulator (return to ground, jump ahead). Each of these keywords is connected to various problems given by the complexity of the flight simulator as whole training machine.

An advantage of using the VLSI technology integrated on a computer board would reside in a better documentation, ease in maintenance and lead to decentralized hardware building blocks. Possibly the flight simulator of the future will have several microcomputers running together, each of them being dedicated to a specific task like the example shown in this paper.

Acknowledgements

I wish to thank C.Houtermans, P. Marti, H-R. Aeschlimann and P. Hoffmann of Swissair for their continuing support of my work and for providing me with valuable data and information.

I would like to thank C. Baker of the Geneva office of General Electric and P. Ackermans of CAE Industies in Montreal for providing me with data of the CF-6/50 engine.

I also thank my friend P. Grepper for his continued support and encouragement during the course of this work.

References :

[1] Intel Corporation: MCS 86 Family Manual and related literature.

[2] F.E. Huguenin: Educational Aspects of Multi-Microprocessor Design Used in Flight Simulation Applications, AIAA Aircraft Systems Meeting. Paper 80-1852

[3] P.O. Grepper/ F.E.Huguenin :4-D Helical approach for transport aircraft in an ATC environment. AAIA Guidance and Control Conference 1979, Boulder Co.

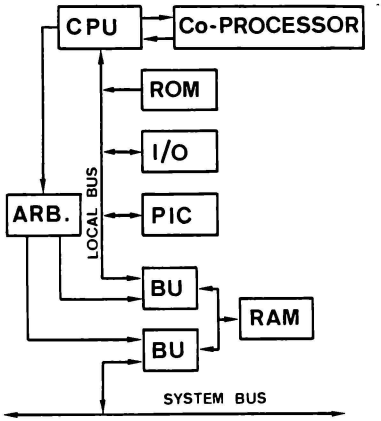


Fig.1 Single Board Computer Hardware

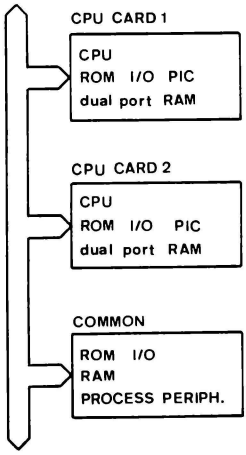


Fig.2 Dual Processor Configuration

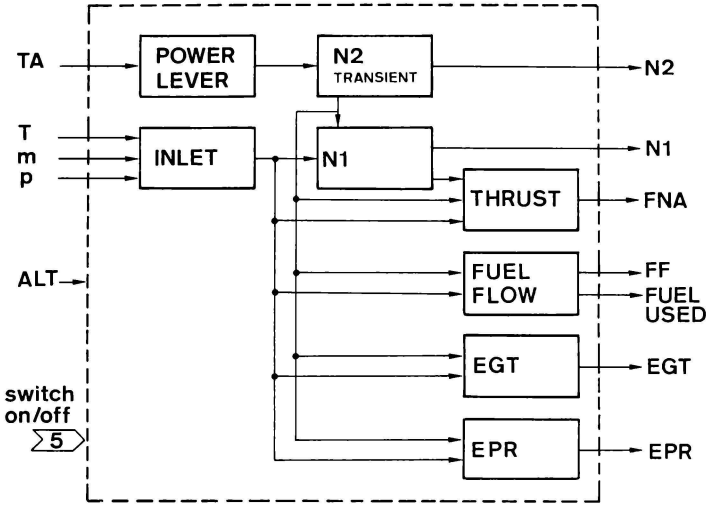


Fig.3 Blockdiagram of the Engine Model

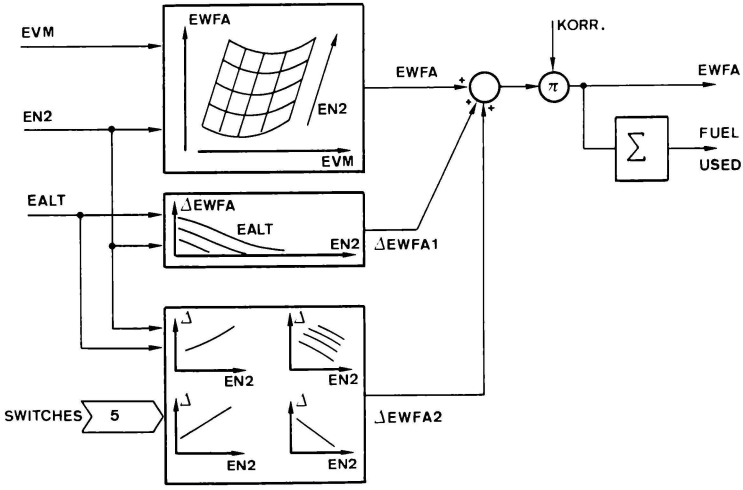


Fig.4 Fuel Flow Module

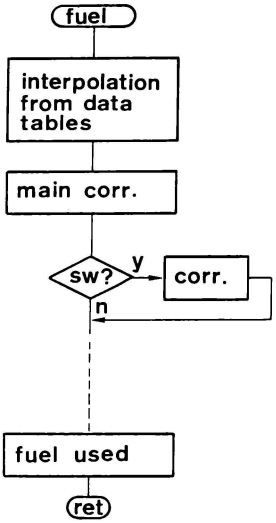


Fig.5 Basic Flow Chart for Fuel Flow Module

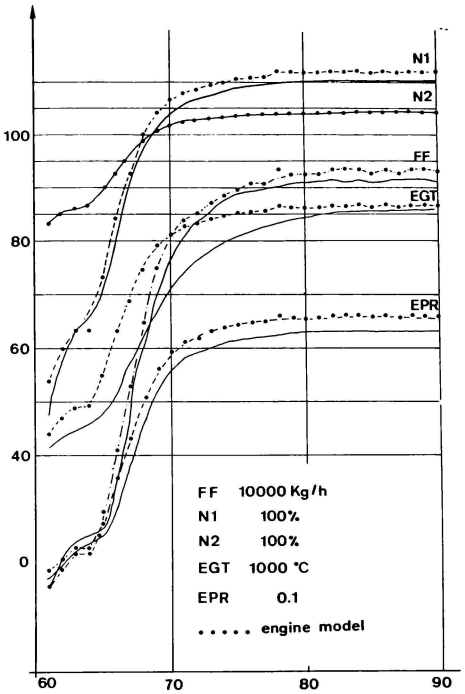


Fig.7 Comparision between AIDS Data and the Engine Model

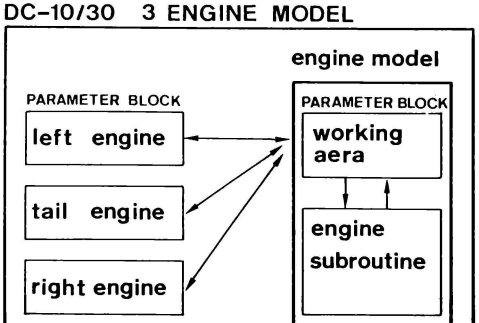


Fig.6 Structure of the 3 Engine Model of a DC-10/30 aircraft

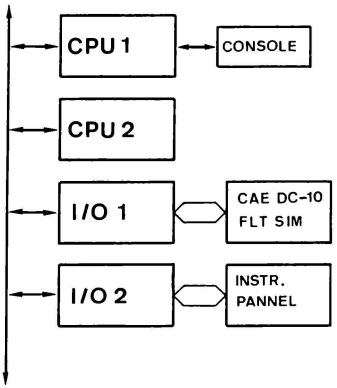


Fig.8 Dual Processor configuration for connecting the Engine Model to a DC-10/30 Flight Simulator

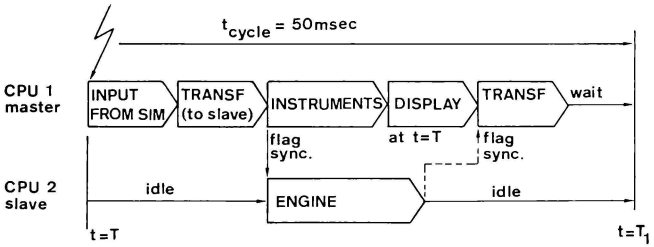


Fig.9 Timing Diagram for the two processor configuration.

81-0974

VERSATILE AND ECONOMICAL REAL-TIME SIMULATION FOR DIGITAL FLIGHT CONTROL SYSTEMS

J. W. Benson, D. B. Mulcare, J. B. Hoenes

Lockheed Georgia Company, Marietta, Georgia

ABSTRACT

An exceptionally efficient real-time airplane model is implemented on a PDP 11/60 and can be used for either analytical or system simulation. The six degree of freedom discrete state model, which iterates the dynamics at up to 100 HZ. and a quasi static trim model are mechanized such that transitioning from one flight case to another is possible. Aside from the state model, the efficiency of the simulation is due to the organization of the simulation software and the utilization of PDP 11/60 operating system.

INTRODUCTION

As real-time aircraft simulations have converted from analog to digital implementations during the past decade, the tendency has been to utilize various integration algorithms to solve the differential equation set representing the aircraft dynamics. This approach to the simulation problem, while very efficient on an analog computer, is highly inefficient on a digital computer. The very nature of the concept of numerical integration applied to the solution of a set of lightly damped, highly coupled differential equations leads to problems in computation time and stability. An analysis of stability charts where the numerical or Z plane roots for a second order system are mapped onto the continuous or S plane for various integration operators indicates a major problem utilizing numerical integration in real-time simulation. The simpler integration methods such as Euler and Rectangular have narrow stability boundaries and are inaccurate due to root shifting unless the sample frequencies are quite high. More complex integration algorithms such as the various Runge Kutta methods are accurate and have wide stability boundaries but are hard to start and consume excessive computational time. Multi-step methods are usually accurate but have narrow stability boundaries and still require excessive time. The disadvantages of utilizing numerical integration for the solution of real-time dynamic models and the inherent efficiency of a digital computer in handling arrays indicates that some form of state model might have computational advantages in the solution of real-time dynamic systems.

State variable methods have been in use for several decades; however, there has apparently been little effort to utilize these methods for real-time pilot in the loop flight simulation. The solution to the dynamic system consists of a series of matrix-vector products and vector summations, two operations that can be made extremely time efficient on a digital computer. The determination of the coefficients of the state transition matrix and the forcing matrix for a complex system can be involved, requiring the exponentiation and inversion of a large state matrix. However, these operations are non-real time tasks, and are calculated in an initialization or background mode.

At any instant in time, the state model yields a perturbated solution to the dynamic system and the state variables are actually incremental about a trim condition. Since purely incremental solutions for piloted simulation would be unsatisfactory, a static trim model was developed so that aircraft trim could be varied in a real-time sense as the pilot responded to changing environmental conditions. The dynamic state solution was then superimposed on the static trim to yield the total system variables required to describe the maneuvering aircraft, and establish a ground reference for such flight control functions as VOR, Localizer, Glide Slope and Automatic Landing.

THE AERODYNAMIC MODEL

The dynamics of a six-degree of freedom rigid body airframe can be described by three-body axis force equations which determine the linear accelerations along each axis, and three-body axis moment equations which determine the angular accelerations about each axis.

$$m[\dot{u} + qw - rv] = T\cos(e) - Wt.\sin\theta + X_B$$

$$m[\dot{v} + ru - pw] = Wt.\sin\phi\ \cos\theta + Y_B$$

$$m[\dot{w} + pv - qu] = Wt.\cos\phi\ \cos\theta - T\sin(e) + Z_B$$

$$Ixx\,\dot{p} - Ixz\,\dot{r} - qr(Izz - Iyy) - Ixz\,pq = L_B$$

$$Iyy\,\dot{q} + pr(Ixx - Izz) + (p^2 - r^2)\,Ixz = M_B + Tze$$

$$Izz\,\dot{r} - Ixz\,\dot{p} + pq\,(Iyy - Ixx) + Ixz\,qr = N_B$$

u, v and w are linear accelerations; p, q and r are angular accelerations; X, Y, and Z are aerodynamic forces; and L, M, and N are aerodynamic moments about the longitudinal, lateral, and vertical body axes of the aircraft.

Given this set of force and moment equations, there is nothing unique about the choice of states to represent the system. However, for efficiency they should be the minimum number necessary to completely define the dynamics and position of the aircraft. All other variables should be defined in terms of the state variables selected. In the case of a six-degree of freedom aircraft it appears advantageous to define an eight variable state vector of the form:

$$[\Delta u, \Delta w, q, \Delta\theta, \Delta v, r, p, \Delta\phi]^T$$

where $\Delta\theta$ is the incremental pitch angle and $\Delta\phi$ is incremental roll angle. θ and ϕ are standard Euler angles describing the attitude of the aircraft with respect to an earth reference.

A partial differential equation can now be written for each of the aerodynamic forces and moments with respect to the state vector and the time derivative of the state vector. These aerodynamic forces and moments are of the general form:

$$F = \frac{\partial F}{\partial u}\Delta u + \frac{\partial F}{\partial w}\Delta w + \frac{\partial F}{\partial q}q + \frac{\partial F}{\partial \theta}\Delta\theta + \frac{\partial F}{\partial v}\Delta v + \frac{\partial F}{\partial r}r + \frac{\partial F}{\partial p}p + \frac{\partial F}{\partial \phi}\Delta\phi + \frac{\partial F}{\partial \dot{u}}\dot{u} + \frac{\partial F}{\partial \dot{w}}\dot{w} + \frac{\partial F}{\partial \dot{q}}\dot{q} + \frac{\partial F}{\partial \dot{v}}\dot{v} + \frac{\partial F}{\partial \dot{r}}\dot{r} + \frac{\partial F}{\partial \dot{p}}\dot{p} + F_{trim}$$

where ($\dot{}$) indicates a derivative with respect to time.

Each of these partial derivatives can be obtained by defining

$$F = QRC_F$$

where Q is the dynamic pressure acting on the airplane, R is the wing area for a force, the wing area times the mean aerodynamic chord for a moment about the lateral axis, the wing area times the wing span for a moment about either the longitudinal or vertical axis,

C_F is the total aerodynamic coefficient with respect to a particular axis force or moment.

The partial derivative can now be expanded into an equation of the form:

$$\frac{\partial F}{\partial \zeta} = \frac{\partial Q}{\partial \zeta} R\, C_{TRIM} + Q\, R \frac{\partial C_F}{\partial \zeta} + Q\, R \frac{\partial C_F}{\partial \xi}\frac{\partial \xi}{\partial \zeta}$$

where ζ represents a state variable and ξ represents a dependent variable such as angle of attack or side slip angle.

The aerodynamic force and moment equations can now be transformed to the body axes and the individual partial derivatives combined into matrices so that a vector equation can be written in terms of the aerodynamic derivatives, state variables, state derivatives, and inertial terms of the aircraft force and moment equations. This vector equation has the form:

$$\dot{\underline{X}} = [Fx]\,\underline{X} + [Fin]\,\underline{X} + [F\dot{x}]\,\dot{\underline{X}} + [Fc]\,\underline{Z} + [D]\,\delta$$

where

$\underline{X}$ and $\underline{\delta}$ are the state and forcing vectors, and $\underline{Z}$ is a vector of the products of the angular velocity terms found in the airframe moment equations.

$[F_x]$ is a matrix of the aerodynamic derivatives with respect to the state vector.

$[F_{IN}]$ is a matrix of the inertial derivatives with respect to the state vector.

$[F_{\dot{x}}]$ is a matrix of the aerodynamic derivatives with respect to the derivative of the state vector.

$[F_C]$ is a matrix of the products of Inertia.

[D] is a matrix of the aerodynamic derivatives with respect to the forcing vector.

[I] is the identity matrix.

The forcing vector $\underline{\delta}$ normally consists of elevator, aileron and rudder deflections and any additional controls which affect the dynamics of the aircraft.

Applying elementary matrix operations, the aircraft vector equation can be reduced to the continuous state form:

$$\dot{\underline{X}} = [A]\,\underline{X} + [B]\,\underline{\delta} + [C]\,Z$$

$$[A] = [I - F\dot{x}]^{-1}\,[Fx + Fin]$$

$$[B] = [I - F\dot{x}]^{-1}\,[D]$$

$$[C] = [I - F\dot{x}]^{-1}\,[Fc]$$

DISCRETE STATE MODEL

The direct solution of the continuous state form would not be advantageous since integration algorithms are required and would therefore defeat the purpose of the state model. To utilize the advantages of digital processing, the continuous state equation must be evaluated to give the solution for a linear continuous system with sampled input. The general solution for a linear continuous system is well documented in the literature on state variable methods and takes the form:

$$\underline{X} = e^{A(t-to)}\underline{X}o + \int_{to}^{t} e^{A(t-\tau)}B\underline{U}(\tau)\,d\tau$$

If the input U(Z) is considered as a sampled zero order hold function so that $u(\tau)$ is constant during any sample period T = t-to the equation reduces to the discrete general form:

$$\underline{X}n+1 = [e^{AT}]\,\underline{X}n + [e^{AT} - I]\,A^{-1}\,B\underline{U}n$$

Where e^{AT} is an exponential matrix and is solved as a matrix power series. Since the state matrix A may contain singularities the matrix function $[e^{AT} - I]\,A^{-1}$ can also be solved as a separate power series to avoid problems with the inversion of A.

Applying this discrete form to the aircraft state equation yields the vector equation to be programmed for real-time solution

$$\underline{X}n+1 = [e^{AT}]\,\underline{X}n + [e^{AT}-I]\,[A]^{-1}\,[B]\,\underline{\delta}n$$
$$+ [e^{AT}-I]\,[A]^{-1}\,[C]\,\underline{Z}n$$

For simulation purposes all of the matrix functions can be evaluated in an initialization or non-real time mode on the computer. The real-time problem then reduces to the task of summing three matrix-vector products. The matrix $[e^{AT}]$ is the state transition matrix and for a six degree of freedom aircraft model is dimensioned 8 X 8. However, for practical purposes the aerodynamic coupling between longitudinal and lateral axes on an aircraft can be neglected. $[e^{AT}]$ can then be partitioned into four submatrices with the off diagonal submatrices equal to zero. This partitioning takes the form

$$\begin{bmatrix} \begin{bmatrix} \text{Longitudinal} \\ 4 \times 4 \end{bmatrix} & \begin{bmatrix} 0 \end{bmatrix} \\ \begin{bmatrix} 0 \end{bmatrix} & \begin{bmatrix} \text{Lateral} \\ 4 \times 4 \end{bmatrix} \end{bmatrix}$$

and can be reduced to the solution of separate 4 X 4 matrix-vector products for each axis. This reduction has the immediate effect of halving the computational load for the closed loop portion of the real-time solution. The state Product of Inertia matrix

$$[e^{AT}-I]\,[A]^{-1}[C]$$

is very sparse with predefined element locations; its expansion into an equation set as a function of the vector $\underline{Z}n$ eliminates the necessity of testing for or multiplying by the zero elements and further reduces the computation time involved.

DYNAMIC TRIM MODEL

Since the state model computes a perturbated vector about the aircraft trim, a trim vector is necessary to relate the aircraft to an earth reference. The initial trim vector is obtained during the nonreal-time setup phase as a function of the desired forward velocity and the aerodynamic configuration. Trim is based on the criterion that all linear and angular accelerations, and all angular velocities must be zero.

For either a pilot or a flight control system to fly a predefined flight pattern the aircraft must be capable of changing its trim condition to be realistic. A dynamic trim model was developed to allow the aircraft to change trim conditions as a function of static elevator position and static throttle position. Quasi static values for angle of attach, pitch angle, dynamic pressure and linear velocities are computed implicitly with all accelerations held at zero. This computation can be done in a background mode at a relatively long sample time. Since no integration is involved the solution is not time critical. Only the longitudinal flight mode needs to be considered. In the lateral mode all of the maneuvers such as slipping or turning are dynamic and are referenced to a zero lateral trim.

FUNCTIONAL IMPLEMENTATION

A DEC PDP 11/60 minicomputer was chosen as the simulation processor. The combination of DEC RSX 11-M operating system, FORTRAN IV plus, and the FP11-E floating point processor provide features which allow floating point programs to run in an efficient time critical environment. During the process of generating an RSX11-M system on the PDP 11 a block of memory can be predefined as a fixed common block which is accessible to all programs running on the system. This common block is normally located between the executive and the user programs or tasks (see Figure 1).

For the simulation model, all aerodynamic variables, aerodynamic parameters, matrix coefficients and other variables shared by multiple programs are defined as fixed locations in the common block of memory. Thus many programs can be running sequentially or in real time simultaneously with varying frame times or cycle rates and have access to the results of each others computations.

Figure 2 illustrates the interaction of the various programs through the common block. In the upper half of the diagram the initialization programs calculate I/O scale factors, trim variables, the state, transition and forcing matrices, and other data required by multiple programs. The lower half of the diagram shows the interaction of the real-time programs with the common block and also the I/O block. Thus any program has complete freedom in modifying any variable associated with the simulation. Most of the extensive computation normally encountered with a flight simulation is accomplished in the initialization phase. The real-time phase then consists mainly of a number of matrix products and summations.

RECONFIGURATION

The state approach lends itself to extensive reconfiguration from a base system. Aerodynamic variations due to changes in flaps, dynamic pressure, and ground effects can be readily accommodated by precomputing incremental state and forcing matrices during the initialization phase. If $[A_1]$ is the state matrix of the base aerodynamic model; and $[A_2]$ is the state matrix of the reconfigured aerodynamic model; a new state matrix can then be defined as

$$[e^{AT}] = [e^{A_1T}] + \lambda\,[e^{A_2T} - e^{A_1T}]$$

λ is a normalized variable and $[e^{A_2T} - e^{A_1T}]$ is usually a sparse incremental matrix. The state solution is exact at the end points for λ equal to either zero or one. The inclusion of reconfiguration matrices require very few data points in the common buffer to define a fairly extensive aerodynamic change. In transitioning from a low speed cruise through the approach and into a landing configuration the state matrix could be updated at a lower frequency as a function of flap position without impacting the computational load of the basic state solution. Utilizing the common block the simulation can be performed using a number of fairly simple real-time programs interacting and executing at their most efficient cycle rates. No function has to be computed at a frame rate higher than is necessary for the required simulation fidelity.

FLOATING POINT TIMING

FORTRAN IV PLUS operates with the FP11E floating point processor to provide efficient in-line code for performing floating point multiplication and summation. The average time to fetch two floating point words from memory and store their product back into memory is approximately 15 microseconds. This is close to the same time as would be required for the operations in fixed point. The primary penalty in performing the aircraft simulation in floating point is the time cost in scaling and converting the fixed point I/O. The scaled conversion requires approximately 45 microseconds per variable. Table 1 lists various functions performed in real time with their frequencies and execution time. If 50 frames per second is used for the state model, the total computation time required per frame is approximately 4 milliseconds yielding a 20 per cent duty cycle for the processor.

ASYNCHRONOUS I/O

For the purposes of this simulation asynchronous I/O was required to mate the aircraft model to a redundant flight computer subsystem. A 64-word random access memory (RAM) buffer was installed in the I/O page of the PDP 11/60 to provide input signals as shown in Figure 1. This RAM is loaded externally with signals corresponding to control surface deflections from the flight control subsystem. The signals are then taken from the RAM in integer form with a standard read instruction. In a similar manner the sensor output signals are written in integer form to a first in-first out buffer (FIFO) where they are sent serially through interface equipment to the flight control subsystem. All output appears to the PDP 11 as a memory write. No special I/O routines other than the floating/fixed conversion is required.

SIMULATION PERFORMANCE

Time histories of the open loop aircraft response are shown in Figure 3. In order to excite the state model a step increase in velocity was introduced into the simulation. The simulated aircraft immediately enters an undamped phugoid mode with a natural period of about 54 seconds. Figure 4 shows the same condition with the flight control system closing the loop. The instantaneous airspeed (IAS) Hold mode of the flight control system is engaged. The variables representing airspeed and pitch angle are brought back onto their trim values without oscillation. The phugoid mode has been completely damped out. Similar runs were made for Mach Hold, Attitude Hold, and Altitude Hold with the same type of results.

Additional closed loop performance was obtained by engaging the Heading select and Instrument Land System (ILS) modes of the flight control system. The aircraft was located parallel to a north-south runway, 40,000 feet south and 10,000 feet west of the south end of the runway. Figures 5(a) and (b) illustrate the response of the aircraft as it moves to a selected heading of 60 degrees, engages a localizer beam, and flies along the beam until it engages the glide slope beam and starts a descent. The aircraft rolls smoothly into and out of a bank angle to obtain the desired heading, rolls in the opposite direction to capture the beam and holds the beam as it approaches the end of the runway.

CONCLUSIONS

A six degree of freedom, ground referenced, retrimmable, real-time aircraft simulation can be implemented efficiently on a minicomputer utilizing a dynamic state model superimposed on a quasi static trim model. The trim model and the alteration of aerodynamic data via a reconfiguration strategy permits the transitioning from one flight case to another; e.g., from a level flight approach case to a descent landing one. The simulation includes output sensors, turbulence, ground effects, beam geometry and load factor computation without impacting the dynamics of the state model. Each phase of the simulation operates as an individual program or task, which can be given any desired priority or cycle time depending on the requirements of the external flight control system.

The success of the state model formulation for the real-time computation of the aircraft dynamic performance has particularly attractive potential. Array processors with pipelining techniques are available which are ideally suited to the solution of the matrix operations required for the state model. Decreasing the duty cycle by implementing the state model on a minicomputer utilizing an array processor holds the possibility of expanding the size of the state matrix to include flexible modes and other higher order effects which are presently not considered for real-time simulation. Thus the same basic simulation can eventually be used from the conceptual phase of system engineering through final development and system validation.

References:
Benyon P.R.
A Review of Numerical Methods for Digital Simulation
Simulation November 1968

GUPTA S.C.
Transform and State Variable Methods in Linear Systems
Wiley

RSX11-M System Manuals
Digital Equipment Corporation

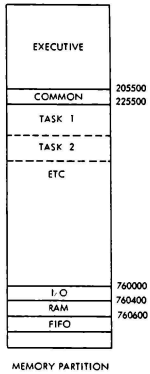


Figure 1.

TABLE 1

Function	Frames/Second	Frame Time (MS)
State Model	50 - 100	2.1
State I/O	50 - 100	.39
Load Factor	50 - 100	.794
Ground Tract	50 - 100	.414
Localizer Angle	10	.182
Glide Slope Angle	10	.186
Angle of Attack	10	.094
Long. Load	10	.078
Radio Altimeter	10	.224
Barometric Altitude	10	.116
Altitude Rate	10	.068
True Airspeed and Mach	10	.190
Yaw Angle	10	.128
Trim Model	1	1.9

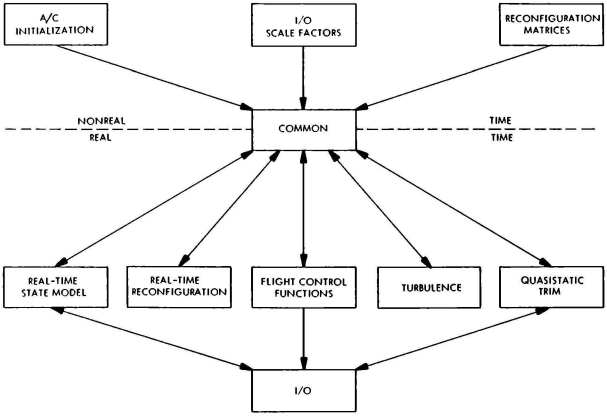


Figure 2

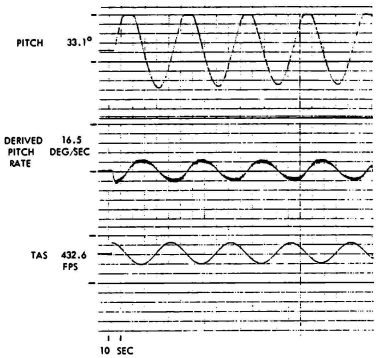


Figure 3.

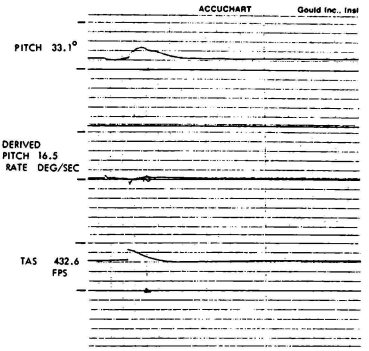


Figure 4.

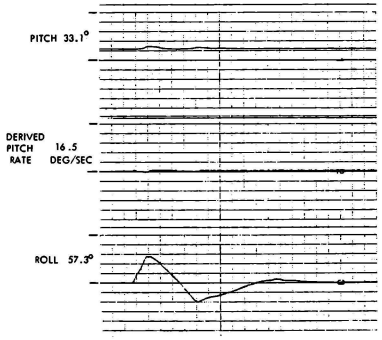


Figure 5a.

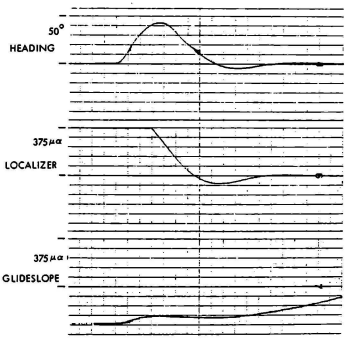


Figure 5b.

DESIGN VERIFICATION BY EMULATION

Dr. Nicholas Szabo
Link Flight Simulation Division - The Singer Company
Sunnyvale, California

Abstract

The cost of either Computer Image Generation or of Digital Radar Landmass Systems is greatly dependent on the required image fidelity. This fidelity can be determined only experimentally. Because the design and construction of a real-time image system can be several million dollars, and because parameters can be varied only by changes in hardware, it is impractical in most cases to evaluate the fidelity requirements on real-time systems. To determine the required image fidelity at reasonable cost, Link established the Image Research Laboratory. This laboratory has the capability to emulate in nonreal time the special purpose image generation hardware and then display sequences of 15-second duration in real time.

Introduction

Over the past ten years great strides have been made in digital image systems for aircrew training applications. Virtually all airlines now use calligraphic, computer-generated images (CGI) to train their pilots. In military and space applications, raster type CGI systems are rapidly replacing camera model visual systems used until recently. Finally, digital radar landmass systems all but obsoleted analog radar image simulators.

The popularity of digital image generation systems is due to their flexibility. An airline pilot, for instance, can within a two-hour training period make a landing at six different cities selected from a data base that contains all the airports of major cities in the world. A comparable capability is obviously economically unfeasible using camera model systems.

Typical CGI systems are highly complex, consisting of special purpose pipeline processors. These processors are used because no general purpose computer has the processing capability to even approach the speeds required. (Typically, calligraphic systems have 5-10 thousand integrated circuits while raster systems may have as many as 50 thousand IC's. The output bandwidth is about 90 megabytes/second.)

Although CGI systems are superior to camera model or film systems in flexibility, they cannot yet, except for night scenes, match the realism of the latter approaches. These deficiencies are due to two limitations of CGI systems: inability to generate sufficient detail, and quantization effects in the digital computational process.

The first problem is likely to be with us for some time in the future; today's CGI systems can typically compute 12,000 edges in a scene, while a real-world image is likely to have detail comparable to many millions of edges. Even with today's rapid advance in microelectronics, we cannot expect to match the real world for many years.

The quantization effects arise because CGI images are inherently sample data systems, both in space and time coordinate systems. Sampling below the frequency of the image or of the relative motion results in aliasing (a scintillation, crawling, double imaging, etc.). (Ref. 1 and 2.)

Since it is unlikely that we will in the near future completely overcome the two major deficiencies of digitally generated images, our goal has to be to minimize the effect of image quality deficiencies on the training value.

Image quality is not a mathematically definable quantity, and the only way to evaluate necessary compromises is to compare several different approaches and have observers rate them. The cost and elapsed time required to actually build several alternative real-time CGI systems is, however, prohibitive. To remedy this, Link built the Image Research Laboratory. This facility has the capability to generate in nonreal time images which are then stored, a frame at a time, on a video disc. The sequence of frames can then be played back at the normal rate to portray a moving scene.

The need for moving (or dynamic) scenes arises from the fact that the eye requires less detail in a rapidly moving scene than in a stationary one, but on the other hand is much more distracted by dynamic aliasing than a static one.

The Image Research Laboratory

The laboratory was established approximately five years ago. At present it employs two general purpose computers and several image storage devices (see Figure 1). The radar and the visual emulation share the same computers but the storage and display devices differ.

Visual System Emulator

Two general purpose computers, a Perkin Elmer 3241 and a Harris 6024, are used to do the emulation of the special purpose hardware. The former runs in multitask mode while the latter has a batch operating system. The emulation is in Fortran, down to the register level. It takes into effect word length and round-off error, but not timing. The image is computed a scanline at a time, and for each pixel the red, green and blue (RG&B) components are calculated to an accuracy of 8 bits, where the 256 values are distributed logarithmically. A 525-line solid-state frame buffer of 24 bits per pixel is used to store a single frame in digital form. The video is generated for each color from the 8-bit output word by using a RAM table-lookup module which has a 12-bit output and drives a 12-bit D/A.

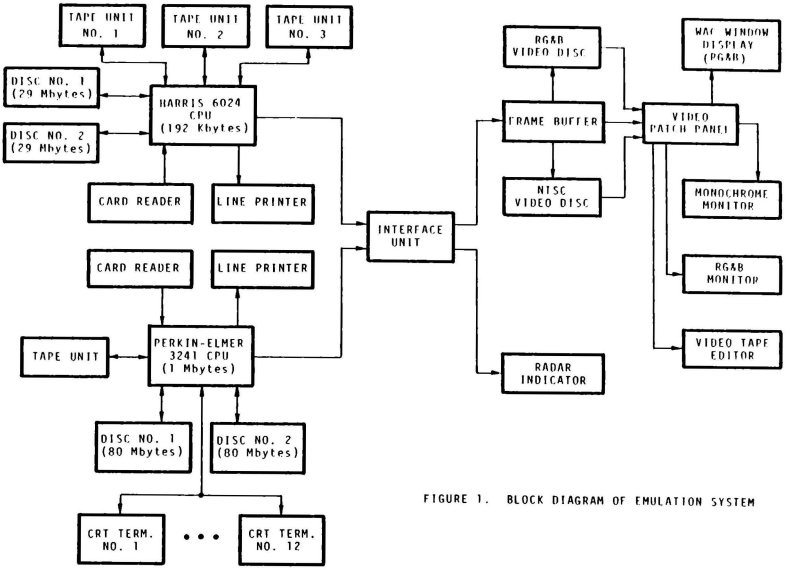


FIGURE 1. BLOCK DIAGRAM OF EMULATION SYSTEM

This table allows varying the video-to-light intensity transfer function. In order to insure proper calibration, once a month the luminous intensity of the display is measured for each color to insure that the 256 different luminous steps are logarithmically distributed. If variations are found, the table can be changed in the RAM. The image is displayed on a monitor with RG&B inputs.

The static image from the solid-state frame buffer can be transferred a frame at a time to either of two video discs. The first one, manufactured by IPS of Belmont, California, has 450 tracks and uses a hard disc. At 30 frames per second it can store either 5 seconds of real-time video in the RG&B mode or 15 seconds of monochrome or NTSC color. The disc has 6 heads, on two independently movable arms. On each arm there is one head for each color. While one arm is stationary and either reading or writing, the other arm is stepped forward to be in position to read the next track.

The ability to generate dynamic images in true RG&B components is important because, with the NTSC format, the spatial frequency of the chrome information is much below the luminance frequency. Since CGI systems do not use NTSC encoding, emulations of color effects must be in the RG&B format.

The second disc is a flexible kind (floppy) and can store about 460 frames but only in monochrome or NTSC color. An NTSC encoder is used to generate the color video for input to the Arvin-Echo disc.

In many applications, scenes of 15 and even 5 seconds in length are adequate. Typically, this may be the case for viewing aliasing. Staircasing along the horizon is best observed by slowly rolling the aircraft from -10° to +10°. A dynamic scene that repeats itself every 5 seconds in which the aircraft is rolled back and forth slowly can be observed by commanding the disc to continuously replay the same 5-second interval. There is no visible interruption at the end of the 5-second period because the video disc heads can completely reposition themselves in a single frame time.

In some applications, of course, longer sequences are required. These can be made by using a SONY Model 2860A Video Editor to splice together 15-second sequences derived from the discs. These long sequences are, however, restricted to monochrome or NTSC color since at present there are no commercially available RG&B magnetic tape recorders.

The video tape is also used as a laboratory notebook for all experiments. The alphanumeric generator of the frame buffer allows a short explanation of the scene to be inserted ahead of the emulated scene.

In recognition of the fact that aliasing is highly dependent on intensity level and the angular subtend of a pixel, a wide angle collimated (WAC) window is used for the viewing of the monitor. This window has a field of view of about 36°X48° and places the image optically at infinity. Since

ordinary CGI displays use 1000-line rasters, the WAC window in the laboratory has a field of view of approximately half of the normal one to compensate for the ability to display only 525 lines.

The emulation of the special purpose CGI processor is done in Fortran and is obviously performed much slower than in the real hardware. Computation times of several minutes per frame are not unusual. The software for the image generation process is highly modular so that, as an example, a new clipping algorithm can replace another one without requiring replacement of the entire emulation software.

Radar Emulation

The scan rate of radar is much slower than the frame time of visual raster generators. For this reason no frame buffers or video tape systems are employed for radar. In most cases static images are satisfactory. Several images in sequence may be stored on the discs of the computer systems.

The radar display employed is a 9" random deflection system that can be programmed for PPI display mode. Offset scan may be employed and the display system's intensity transfer function may be varied by using a RAM lookup table. A high persistence radar phosphor is used on the 9" tube. The limiting resolution of the display is 200 lines/inch but the spot can be defocused to simulate lower resolution radars. The deflection system of the display is driven by 14-bit D/A's in both the X and Y directions. By incrementing both the X and Y inputs, one may generate ramps in any direction desired. The X and Y increments are loaded by the computer for each scan.

To keep a record of experimental results, photographs are taken of the display.

Many newer radar systems now employ digital scan converters and raster displays. To emulate these, the visual image emulator may be used.

Applications

The Image Research Laboratory has been used extensively to determine image quality. Many experiments were carried out to develop antialiasing algorithms in both monochrome and color. The number of subpixels per pixel and various spatial filters for alias suppression were evaluated, both statically and dynamically. Furthermore, the effect of transparency, various levels of detail and transitioning from one level to another were emulated. In the area of texture generation, the emulation process was used exhaustively to insure that the computation was carried out with sufficient accuracy. Several patterns to resemble wave motion on the water and natural vegetation were developed. A great deal was also learned in the art of suppressing scintillation in texture patterns (see Figure 2).

In the area of radar images, new implementations for beamspread, aspect angle and clutter effects were developed (see Figure 3).

Figure 2. Emulated CIG Image of Texture.

Figure 3. Emulation of Radar Image Showing Experiment in Beamspread.

Conclusions

Overall, it is safe to say that to develop the same algorithms and to test them in real-time hardware would cost a factor of 30 times as much and would have required four times longer. Based on the success of emulation, Link has adopted the policy of requiring complete functional emulation of any major design change. In some instances emulation can be accomplished during the proposal phase to reduce risks while in others emulation is used from feasibility studies to design verification of contracts already awarded.

References

1. Szabo, N. S. Digital Image Anomalies: Static and Dynamic. Visual Simulation and Image Realism, Proc. 22nd Annual International Technical Symposium, Society of Photo-Optical Instrumentation Engineers. San Diego, Calif., (30-31 August 1978), 162, 11-15.

2. Lipscomb, James S. Reverse Apparent Movement and Erratic Motion with Many Refreshes per Update to appear in ACM Computer Graphics.

81-0977

SPACE SHUTTLE DYNAMIC INTEGRATED TEST: CONCEPT AND RESULTS FOR STS-1

Steven Brody*
Intermetrics, Inc.
Huntington Beach, CA

Richard W. Weissberg
Intermetrics, Inc.
Cambridge, MA

ABSTRACT

The Dynamic Integrated Test (DIT) allows a simulation of launch and flight scenarios to be exercised using the flight-ready Space Shuttle vehicle at the Kennedy Space Center. The purpose of this test is to provide a critical verification of the integrity of the assembled vehicle and the launch support environment. In essence, the DIT causes the shuttle to believe that it is flying, so that hardware and software systems are exercised much as they would be for a real flight. The DIT technique is a specific application of the ASIST concept (Avionics System Integration Self Test) developed by Intermetrics.

1 INTRODUCTION

Avionics systems have evolved to such a level of complexity and expense that it has become impractical in many cases to perform 'throw away' or high risk flight tests. Man-rated systems most often must fly the first time out with a crew complement. Hence, it is incumbent upon the systems engineer to consider as a critical component the requirement to adequately ground test a vehicle.

Typical ground testing involves end-to-end checks of various subsystems often by stimulating an input device and examining the resultant effect. The movement of a cockpit control stick, for example, may be performed to examine proper interaction with the ailerons. These tests, however, do not exercise dynamically all subsystems interactions encountered during a real flight. A new concept of Avionics System Integration Self Test (ASIST) has been developed here to provide just such an overall systems test.

This paper examines the development and implementation of the ASIST concept for NASA's Space Shuttle, officially entitled Dynamic Integrated Test (DIT). The performance of DIT prior to the maiden voyage of Columbia yielded significant return by instilling confidence in the readiness of the vehicle and the launch support environment.

1.1 The Space Transportation System

The Space Transportation System (STS) is planned to be the workhorse of NASA and DOD in the 1980's and beyond. Illustrated in Figure 1, it consists of the Space Shuttle Orbiter spacecraft with its three main engines, an external tank housing the main engine propellants, and two solid rocket boosters. The Space Division of Rockwell International is prime contractor to NASA for the Space Shuttle Orbiter as well as for total integration of the STS. At present, the orbiter spacecraft containing the crew and payload bay is the only component that reaches orbit. After completing its mission, the shuttle then returns to earth, landing as an airplane.

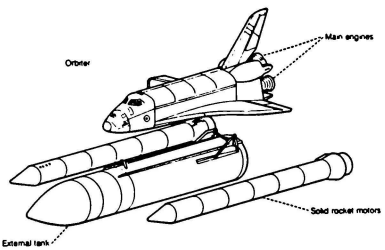


FIGURE 1

Configuration of the
Space Transportation System

* Member AIAA

Central to the operation of the STS are the five on-board IBM AP-101 flight computers which monitor and control the vehicle's flight operations. In fact, the shuttle has been designed as a digitally controlled spacecraft; that is, there are no direct mechanical linkages by which the crew can manipulate the aerosurfaces or cause the various propulsion subsystems to fire. The vehicle is unstable in most flight modes and generally requires continuous control by the computers to maintain stability within rigid boundaries. In certain flight modes the crew does have the option to select either the automatic mode for guidance, navigation, and control (GN&C) of the vehicle, which essentially allows the computers to do all the flying, or the manual mode of flight, whereby the crew flies the vehicle by hands-on manipulation of the controls. However, if manual flight is selected, the commands issued by the crew to the aerosurfaces and the engines must still pass through and be issued by the computers. Hence, the reliance on the computers is absolute.

The five flight computers, called general purpose computers or GPCs, are organized into a redundant set of four GPCs which form the primary flight system (PFS) plus a single GPC used as the backup flight system (BFS). IBM has responsibility not only for the flight computer hardware but also for the software which runs in the four PFS GPCs. This code, written by IBM to implement Rockwell specified requirements, is loaded identically into each of the four redundant computers. During flight operations, inter-computer communication insures self-consistency and synchronization within the PFS.

To provide an additional level of redundancy, a fifth GPC operates independently of the PFS GPCs, executing software developed and coded by personnel from Rockwell, Intermetrics, Inc., and the Charles Stark Draper Laboratory. This backup flight software satisfies requirements established by Rockwell similar to those for the PFS. The crew engages the backup flight computer in the event that critical failures are detected in the PFS. The primary and backup flight software are both written in HAL/S, a high-order structured programming language developed and supported by Intermetrics, Inc. for NASA.

Figure 2 depicts the essential components of the orbiter avionics system. The central computers interface with the various subsystems through 19 serial data buses. Eight of these connect with the flight forward and flight aft multiplexer/demultiplexer (MDM) units

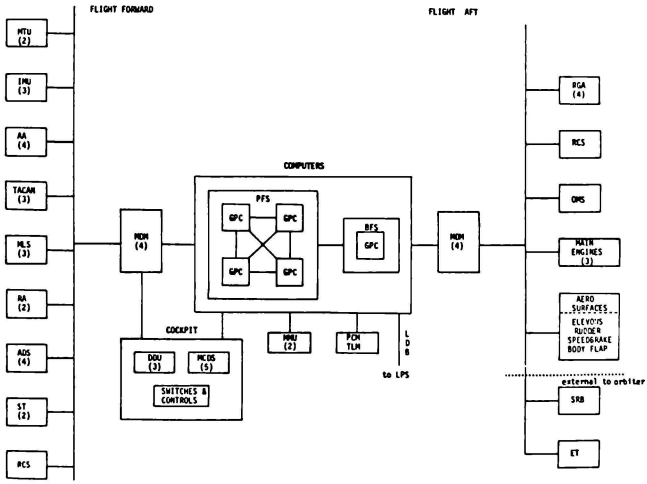


FIGURE 2 Orbiter Avionics Subsystems

which in turn serve as the conduit for signals going to and from such devices as 1) the master timing unit (MTU) which serves as the accurate time source for the GPCs; 2) sensors which provide velocity and attitude information such as the inertial measurement unit (IMU), the accelerometer assembly (AA), and the rate gyro assembly (RGA); 3) external navigation aids ("navaids") such as the tactical air navigation system (TACAN), microwave landing system (MLS), radar altimeter (RA), air data system (ADS), and star tracker (ST); 4) propulsion subsystems such as the reaction control system (RCS), the orbital maneuvering system (OMS), and the main engines; 5) the vehicle aerosurfaces such as elevons, rudder, speedbrake, and body flap; and 6) the cockpit switches and controls and dedicated display units (DDU) which provide to the crew information such as shuttle velocity, altitude, attitude, and other flight critical parameters.

Other data buses connect the GPCs with the multi-function CRT display systems (MCDS) which present a variety of GN&C and system status information to the crew, and allow them extensive interaction with the GPCs via keyboard commands. The mass memory units (MMU), which contain the primary and backup flight software to be loaded into the computers as needed, also communicate directly with the GPCs. Another data bus is a port through which the GPCs transmit pulse code modulation (PCM) telemetry (TLM) data used by the ground to monitor the internal state of the GPCs and the progress of the flight, either in real-time or after the mission.

Finally, the launch processing system (LPS) which interfaces with the vehicle prior to launch does so via the umbilical launch data bus (LDB), which allows direct access to GPC memory. In addition to performing vital ground launch sequencing tasks, the LPS, using a general memory (GMEM) read/write capability, can monitor and modify GPC core, if needed; for example, to update launch pad initial conditions. As will be seen later, this capability is essential in order to perform a dynamic integrated test (DIT) of the shuttle.

The STS is designed to follow certain basic flight sequences from launch to landing. Abort contingencies have been considered as well and are included in the design of the software to allow for the orderly and safe return of the orbiter spacecraft. These contingencies are 1) return to launch site (RTLS), which would be performed in the event the mission had to be aborted soon after launch; 2) abort to orbit (ATO) in which the orbiter spacecraft is sufficiently healthy to allow insertion into earth orbit; and 3) abort once around (AOA) where the vehicle is boosted to permit one orbital revolution before landing.

During a nominal ascent, the solid rocket boosters (SRB) and main propulsion system fire together until the SRBs are depleted and jettisoned after approximately two minutes. The fuel in the external tank is exhausted after a total burn time of about eight minutes, the main engines are shut down, and the external tank is separated. The orbital maneuvering system (OMS) is then used to execute orbit insertion and to circularize the orbit. OMS firings occur at times which may vary with the mission profile.

After earth orbit operations are complete, the OMS is once again used for the deorbit burn to begin a nominal descent. The orbiter coasts for about 30 minutes to atmospheric entry. Approximately another 30 minutes places the shuttle on the runway having transitioned through several guidance phases and performed the requisite landing maneuvers. The shuttle makes a dead-stick landing as a glider; it was not designed to allow a go-around cabability.

1.2 Motivation Behind the DIT

The various subsystems -- computers, flight software, navigation instruments, environment sensors, engines, etc. -- are subjected to rigorous checkout and verification procedures, at least to the fullest extent possible in a testbed environment. Then they are shipped to Kennedy Space Center (KSC), Florida, for final integration into the shuttle vehicle. But it is not until these subsystems are received and fitted to the vehicle at KSC that the Space Transportation System exists which will carry a crew into orbit. Clearly, just as checkout and verification of each of the component subsystems is vital, so is it necessary to verify the final integrated configuration. The complete hardware-software data paths which are utilized in flight simply do not exist until the vehicle is put together. The integrated test is necessary to verify the integrity of these paths: that proper connections have been made, that correct polarity is maintained, and that no unexpected interference is generated. The concept of the integrated test is complicated by the fact that the vehicle is not static -- the software and hardware configuration changes during the course of a mission as the software is moded through various sequences. Thus a valid integrated test is necessarily a

dynamic integrated test, which checks out hardware-software data paths in a vehicle which is sequenced through the events of a flight profile.

1.3 A Brief Overview

A mechanism conceived by Intermetrics, Inc., has been developed for realizing this DIT concept. In essence, a simulated flight sensor profile is transmitted over the umbilical launch data bus and injected into the flight software in such a manner that the grounded vehicle believes it is flying. The vehicle is sequenced through nominal flight phases and deceived to detect the environment, position, velocity, and attitude it would detect on a nominal mission.

In fact, it is not the hardware sensors which are stimulated; the sensors do detect their earth-fixed environment, or in some cases, the constant output of a test set. Rather it is locations in the software which are changed by the DIT. Instead of accessing locations written to by the sensors, the shuttle navigation software accesses locations containing injected simulated data. The navigation software thus runs open loop, in the sense that it is the "canned profile" which is "flown," and flight control commands cannot affect this profile as they would in real flight.

It is the intent of the DIT to exercise as much as possible the complete hardware-software data paths from sensors to effectors. Toward this end, the DIT design permits a choice between two similar but somewhat different data injection techniques for each sensor. Nominal data flow from sensors through applications programs to effectors is schematically represented in Figure 3a. The two data injection modes, known as substitution and combining, are illustrated in Figures 3b and 3c. In the substitution mode, the real sensor data are processed by the input routines but do not influence the GN&C applications calculations nor the commands sent to the effectors. In the combining mode, the real sensor data are added to the simulated sensor data, and it is the resultant combined values which drive the applications routines and effectors. The combining mode allows a more complete verification of the data paths during a DIT, but does involve added complications which are discussed later.

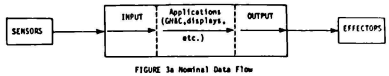


FIGURE 3a Nominal Data Flow

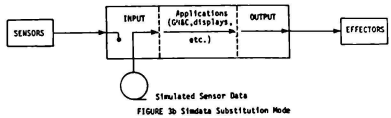


FIGURE 3b Simdata Substitution Mode

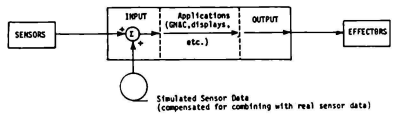


FIGURE 3c Simdata Combining Mode

A few selected nominal profiles are chosen to be "canned" for the DIT; no attempt is made at exhaustive testing of all possible profiles. The selected nominal trajectories serve to mode the software through its various sequences and facilitate verification of hardware-software interfaces and interactions. The various aerosurfaces, engine gimbals, and other effectors on the vehicle are moved (except where prohibited for safety reasons) much as they would be for a real flight.

The rest of this paper examines in some detail the design and implementation of the dynamic integrated test for the Space Shuttle. The ground rules and constraints upon which the design was formulated are presented in Section 2. The concept of the DIT is expanded in Section 3, along with a discussion of the simulated sensor data (simdata), how it is obtained, and how it is transmitted to the GPCs for processing during a DIT. In Section 4, the interaction of the simdata with the flight software is examined, and the software modifications required to implement the DIT concept are presented. Concerns of vehicle and personnel safety during a DIT are discussed in Section 5, along with the techniques developed to ensure a safe checkout of the STS. Section 6 is about test evaluation. The results of the tests performed in support of the maiden flight of Columbia, STS-1, are given in Section 7. A discussion of the future of DIT as the Space Shuttle approaches operational use is presented in Section 8, and finally, in Section 9 we examine the application of the DIT/ASIST concept to other avionics systems.

2 GROUND RULES AND CONSTRAINTS

Certain ground rules were established during the initial design phase of DIT, some to satisfy practical constraints imposed by the existing STS design and development plan, and others to insure that the intent of DIT would not be lost during implementation. Included in the latter category is the design goal to minimize the impact DIT has on the flight software, both in GPC memory requirements imposed by DIT modifications to the software, and in additional execution time (i.e., duty-cycle) resulting from these modifications. If the DIT-modified software were not very similar to the flight software, the relevance of the test would be in question. It is important that the test exercise as much real hardware as possible. Concern for vehicle and personnel safety limits the ability to fully realize this goal, however. Indeed, safety considerations (discussed later) are a significant aspect of the DIT.

The ability to halt and restart the GPCs at specific points during a test, heavily relied upon during laboratory testing, does not exist once the vehicle is assembled. Control of the flight computers for development and checkout becomes quite restricted, relying on normal crew interface through the cockpit plus the capability provided by the launch data bus (LDB). This limits the ability of the test engineers to interface with the various components of the shuttle. Since DIT must run at KSC after the subsystems have been assembled to form a flight-ready vehicle, DIT must operate under the constraint that once it has started, it must complete without interruption.

Certain restrictions are placed on the use of the LDB. During nominal preflight activity, from nine minutes to five seconds prior to lift-off, the critical ground launch sequence requires a dedicated LDB. Hence, there can be no usage of the LDB by DIT during this interval. Another constraint is that DIT utilize no more than 50 percent of the capacity of the LDB, thus allowing a retry in the event of a transmission failure.

A ground rule established for DIT concerns the preservation of the integrity of redundancy management (RM), the software function which selects from among the redundant sensors the set of variables to be accessed by the applications software. The DIT design allows the RM to execute based on inputs from the real sensor hardware, allowing comparisons and fault detection to take place exactly as in real flight. It is only after RM is executed that DIT-supplied simulated sensor data is injected and accessed by the applications routines.

The DIT design necessarily needed to be flexible enough to support development at Rockwell's Flight Systems Laboratory (FSL) and NASA's Shuttle Avionics Integration Laboratory (SAIL). The availability of shuttle hardware, simulators, and other support equipment varies substantially at these facilities; so the DIT was designed to be as independent of specific equipment as possible.

Finally, it is a design goal to allow as much crew participation as possible, including manual keying in of commands to effect the transitions of the software through the various GN&C flight modes. In the spirit of flexibility, however, it was desired to be able to automatically issue these commands to the software via the LDB-transmitted simdata if a crew were not available, or if it was found that time criticality of certain events precluded manual key-ins. Although this 'auto-crew' function was not implemented for STS-1 DIT tests, techniques for mechanization have been examined. Preflight activities for DIT closely follow the standard items to be followed in a real flight, including full cockpit and subsystems checklists.

3 THE DIT MOVIE

In some ways, the running of a DIT is analogous to the showing of a motion picture. A reel of movie film is a series of snapshots which are correlated linearly with time so as to create a dynamically flowing image when projected. For a DIT, "snapshots" of sensor data are stored on the simdata tape and correlated linearly with time in an analogous manner. As the movie projector projects each snapshot on the screen in timed succession, so does the launch processing system transmit simdata frames over the LDB to be injected into the flight software.

3.1 The Simdata Tape

The simdata tape, a standard nine-track magnetic tape, is the medium on which the DIT canned sensor profile is stored. It contains four files of data: three of initialization data (not discussed here), and a fourth which contains the frames of simulated sensor data -- simdata -- to be dynamically injected into the flight software during execution of the DIT. Although

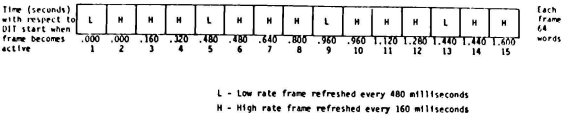


FIGURE 4 Arrangement of Simdata Frames

every frame contains exactly 64 16-bit words, there are actually two types of frames; high-rate, which are refreshed every 160 milliseconds, and low-rate, which are refreshed every 480 milliseconds. Furthermore there are two types of low-rate data, as will be discussed shortly. The frames are arranged on file four of the tape as shown in Figure 4.

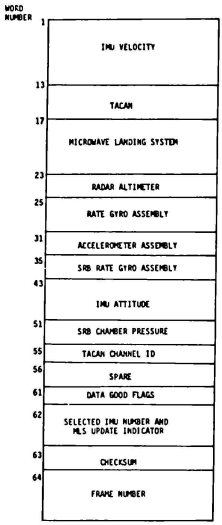


FIGURE 5 Simdata High-rate Frame

The high-rate frame, shown in Figure 5, represents a snapshot of navigation sensor and vehicle status information. Each piece of data has its unique place in the frame template, even though not every sensor is utilized for every DIT. An ascent trajectory requires no tacan or microwave landing system, for example; nor does a descent employ the solid rocket boosters. In any case, the DIT software can uniquely identify each piece of data by its position within the frame, and the data (along with low-rate data, described below) is sufficient to drive the vehicle.

There are two types of low-rate simdata: one which is sensor data refreshed every 480 milliseconds much as high-rate data is refreshed every 160 milliseconds, and a second type which provides a more generalized capability to set the value of any parameter in the GPC memory. As of this writing, the only type-1 data, and thus the only data refreshed in every low-rate frame, is that associated with the air data sensor.

WORD #	CONTENTS
1	Triplet #1 Destination Address
2	Triplet #1 MASK
3	Triplet #1 Data
4	Triplet #2 Destination Address
5	Triplet #2 MASK
6	Triplet #2 Data
7	
8	
9	
•	
•	
•	
•	
•	
49	Triplet #17 Destination Address
50	Triplet #17 MASK
51	Triplet #17 Data
52	Auto-Crew-Indicator = 0
53 54	Selected ADTA Static Pressure
55 56	Selected ADTA Center Pressure
57 58	Selected ADTA Lower Pressure
59 60	Selected ADTA Upper Pressure
61 62	Selected ADTA Total Temperature
63 64	Suncheck Frame Number

FIGURE 6

Simdata Low-Rate Frame

The second type of low-rate data is used to communicate the "occurrence" of various asynchronous events such as detection of weight on landing gear. As shown in Figure 6, each type-2 datum is organized in the low-rate frame as an ordered triplet of 16-bit words: a GPC address, a mask indicating which bits are to be set, and a data word indicating the values of the bits to be set. Up to 17 triplets may be transmitted in a single low-rate frame.

3.2 Obtaining Simdata

A motion picture is made by photographing the real world with a movie camera. An analogous procedure is utilized to generate simdata for a DIT: "snapshots" are taken from the GPC during a closed loop digital simulation in an avionics laboratory.

In facilities such as Rockwell's Flight Systems Laboratory (FSL) and NASA's Shuttle Avionics Integration Laboratory (SAIL), mechanisms have been developed for simulating the shuttle and its environment. As mentioned earlier, the GPC communicates with the world through 19 serial data buses. All sensor inputs and all effector commands are conveyed on these buses. In the laboratories, as part of the flight software verification process (independent of DIT), realistic closed loop simulations are run by connecting these buses to the shuttle and environment models. The software is effectively led to believe that it is flying. (Unfortunately, a similar procedure cannot be performed on a flight-ready vehicle at KSC. Hence the need for a DIT!)

Simdata is obtained from such a simulation by the procedure illustrated in Figure 7. First, before the simulation, a table driven program is implanted in the flight software to transmit data (a maximum of 128 16-bit words) over the GPC telemetry channel every 40 milliseconds. The table contains the addresses of all the data necessary to generate a simdata tape file 4, both high-rate and low-rate frames. The simulation is then run, and the traffic on the telemetry bus is recorded on tape. This "modified telemetry" tape is then processed by another support program known as SIMGEN, which generates a simdata tape.

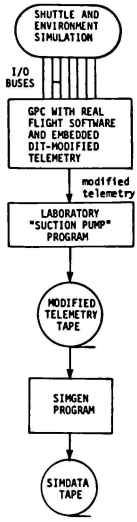


FIGURE 7 Generating a Simdata Tape

3.3 Transmitting the Simdata to the Vehicle

Discussed here is the procedure developed for moving simdata from the tape into the GPC over the launch data bus. Central to the mechanism is that the launch processing system has the capacity to transmit a 128-word data packet to the GPC every 120 milliseconds. This operation, called the GMEM Write, is a standard LPS capability, and is a logical choice for transmitting simdata to the vehicle. In addition to using the GMEM Write operation, these other constraints had to be considered in developing the procedure:

- The DIT should utilize no more than 50 percent of the capacity of the LDB.
- Three high-rate frames and one low-rate frame (each frame 64 words) must be transmitted every 480 milliseconds.
- Data should not be refreshed too early or too late.
- Time synchronization must be maintained among the LPS, the GPC, and the simdata.

The mechanism which evolved from these constraints is depicted in Figures 8 and 9. A DIT patch, really a program in itself known as the GPCCP (GPC Control Program), is embedded in the flight software to receive and process the transmitted simdata. The LPS and the GPCCP each maintain two 128-word buffers, each of which holds two simdata frames; either two high-rate, or one low-rate and one high-rate. The frames are transmitted in pairs over the LDB according to the timing diagram in Figure 9, and double buffering assures that data is not overwritten while it is being read.

Each transmission has a 240 millisecond window which allows for a second attempt to transmit if the first should fail. Normally, if a successful transmission is not made before the window closes, the data which would have been sent is thrown away since it is critical to maintain GMT synchronization. The system is actually quite robust in that it does remain stable through occasional data dropouts; the GPCCP simply turns off the data good flags associated with the various sensors whenever data dropouts occur. The inertial measurement unit (IMU) velocity data transmitted is accumulated velocity, not delta velocity, so even occasional missing IMU data is not catastrophic. The one exceptional case for which the system cannot tolerate a data dropout is when the transmission includes a low-rate frame containing asynchronous data. In fact, most low-rate frames do not contain asynchronous data, but the few frames which do are all critical to a successful DIT. If a transmission window does close on a transmission containing such a frame, further attempts to send the data are made. If combining rather than substitution has been selected for a DIT, the LPS performs a minor transformation on the simdata before transmitting it over the

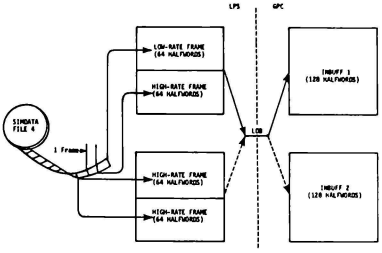


FIGURE 8 DIT Buffers in GPC and LPS

LDB. Since the simdata are added to corresponding real sensor data (representing the earth-fixed environment or test set data) by the GPCCP, the sensor values must be offset appropriately before they are transmitted. The LPS calculates these offsets and performs the necessary subtraction. For most sensors, this offsetting procedure represents a trivial calculation, since the quantity subtracted is constant, representing the static output of a test set. However, the IMU represents a special case, for even in a ground environment its outputs are dynamic, reflecting the rotation of the earth. Thus the LPS DIT control program must incorporate a "ground nominal" IMU model to predict the changing values of IMU gimbal angles and accumulated velocities during the course of a test. When combining is selected for the IMU, these predicted values are generated in real time and subtracted appropriately from the simdata IMU values before transmission over the LDB. As of this writing, the IMU combining option has not been implemented due to the increased cost and development time it implies.

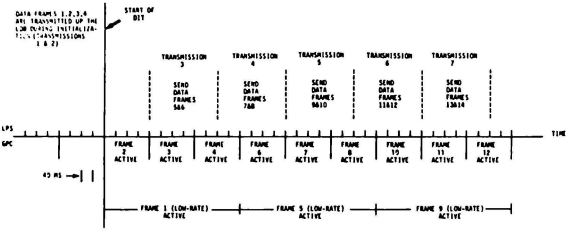


FIGURE 9 Steady State Relationship Between Data Transmission over LDB and Utilization of SIMDATA by GPCCP

3.4 The DIT and Time

The DIT on the vehicle takes place in the time frame of the digital simulation from which the simdata for that DIT was generated. All clocks involved in the test, both in the LPS and on the vehicle, are necessarily initialized so the time of DIT start (see Figure 9) is within a few milliseconds of the corresponding time during the digital simulation. This requirement arises from the fact that time -- GMT -- is one of the inputs which drive the shuttle navigation software. It is, for example, an integral part of the procedure which performs the transformation from the earth-fixed reference frame to the inertial frame used by navigation.

The relaxation of this requirement is anticipated for later DIT usage with the development of the variable DIT start (VDS) technique discussed in Section 8.

4 MODIFYING THE FLIGHT SOFTWARE FOR THE DIT

4.1 Injection of Simdata

The normal (i.e., not modified for DIT) data flow from the sensor subsystem outputs through the MDM's into the flight software is depicted in Figure 10. Associated with each sensor/navaid subsystem is a part of the flight software known as the SOP (subsystem operating program) responsible for transforming the incoming, redundant, raw subsystem outputs into selected, scaled and biased data which can be used by the applications software (e.g., navigation). It is the SOPs which for the DIT are modified to access the simulated trajectory data instead of the ground environment data they would otherwise read.

As mentioned earlier, high-rate simdata is refreshed in the GPC every 160 milliseconds. This is performed by a program (discussed below) known as the GPCCP (GPC Control Program) which executes every 160 milliseconds. The mechanism used to make a SOP access simulated trajectory data is to have the SOP reference a DIT buffer known as SIMBUFF instead of the flight software location normally referenced. It is this buffer the GPCCP refreshes every

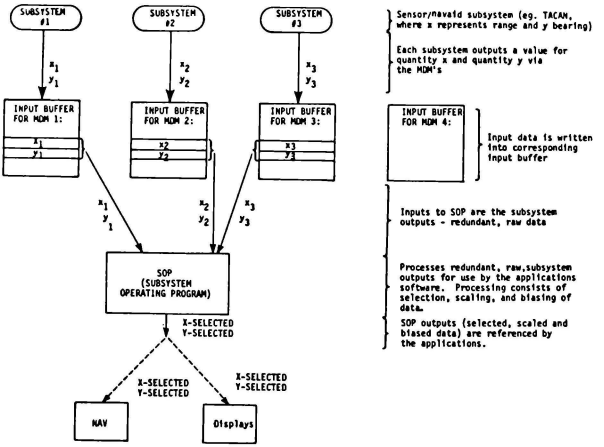


FIGURE 10 Nominal Sensor/Navaid Data Flow

160 milliseconds. It does this by extracting data, which has been transmitted by the LPS, from the appropriate LDB input buffer. The decision on where to make the data injection patch was based primarily on two factors. First, it is much easier to patch a variable to reference simdata at the one place it is set in the SOP than at the several places it is referenced by the applications software. Second, although the shuttle subsystems are redundant sets, because of LDB capacity constraints it is the selected value rather than the redundant values for a given quantity which are injected. In Figure 10, for example, X_SELECTED is injected rather than X1, X2, and X3.

4.2 Other DIT Patches to the Flight Software

Some other flight software patches besides the data injection patches in the SOPs are also necessary for DIT. The largest patch is the GPCCP, which has already been introduced. In addition to controlling the timely flow of high-rate data from the LDB input buffers into SIMBUFF, it also processes the low-rate frame data. The GPCCP contains logic as well to detect LDB transmission failures and in such a case sets invalid data indicators.

Another set of patches is necessary for initialization of the DIT. For example, a patch is needed to start the GPCCP at the appropriate GMT. For ascent, a patch is necessary to delay the start of navigation by four seconds to prevent navigation from executing before the simdata which drives it has begun to arrive over the LDB. (The LDB is not available to DIT before T minus four seconds.) Data as well as code needs to be patched: initialization data such as simulation launch site co-ordinates have to be properly set.

For a descent DIT, initialization is more complex than for ascent, because normally a flight begins on the ground. The descent DIT is initialized by overwriting the flight software GN&C common data base so that when the descent software is started on the ground, it believes the vehicle is in orbit, having just flown a nominal ascent trajectory. The process is further complicated by the fact that in order to exactly synchronize the position, velocity, and attitude with GMT, the navigation and attitude states must be set to the simulation initial conditions and "frozen" until the moment of DIT start. Once the DIT starts, however, the navigation and attitude states are propagated normally, based on simdata.

Still another set of patches is concerned with flight sequences such as solid rocket booster (SRB) ignition, SRB separation, main engine cutoff, external tank separation, and orbital insertion. Each of these sequences involves critically timed interactions with various external subsystems such as the main engines or external tank. Since actual firing of pyros is infeasible for DIT, simulators must be substituted for subsystems whose operation would be hazardous. If a simulator is not available, the software must be modified to sequence correctly without feedbacks it would normally need. However, these DIT sequence patches are designed, in the spirit of the ground rules discussed earlier, such that if the necessary feedbacks are in fact available, the patches can be "turned off" to allow the software to function as it normally would. These patches especially facilitate DIT development in the laboratory, where simulators frequently are not available.

The patches discussed thus far represent (with the exception of inhibits of hazardous outputs, discussed in Section 5) the categories in which most DIT patches may be classified: data injection, GPCCP, initialization, and sequencing. In addition, there are various miscellaneous patches which do not fit conveniently into any of these categories. One example is a patch to allow LDB processing to continue beyond SRB ignition, the point at which it is normally terminated during a real flight. Another example is a patch to cause the cockpit displays, some of which would indicate an earth fixed environment during a DIT, to reflect the simdata instead. A final example is a patch which allows the time-consuming IMU gyrocompass alignment procedure to be bypassed in order to facilitate the development process in the laboratory (this patch is not employed on the vehicle, however).

5 VEHICLE AND PERSONNEL SAFETY

The desire to have DIT emulate a flight must be weighed against the possibility of events occurring which might damage the vehicle or endanger personnel. The aerosurfaces must not be commanded to move so they strike obstructions such as test stands or fuel lines which may be a part of the ground environment. Certainly DIT testing must not cause the engines to fire! Equally catastrophic would be to allow the solid rocket boosters (SRB) or external tank to separate from the orbiter as they do during ascent.

As a result of these concerns, an inhibit methodology using a twofold approach was adopted. First, whenever possible, the hardware action which would be hazardous if permitted to occur is inhibited by physical means such as disabling a circuit via a circuit breaker or by not installing a pyrotechnic device. Second, a software technique is used to prevent hazardous commands from even being issued to the hardware. This software method may consist of either intercepting and nulling a command, as in the case of an engine fire command, or reducing a command to a safe level. An example of the latter case is the reduction of a rudder movement command from 10 degrees to five degrees if allowing the full rudder travel would cause it to hit an obstruction.

The software inhibit technique has been developed so the test conductors can select, prior to a DIT, those commands which must be inhibited, choosing from a set of possible hazardous commands established during DIT development. (In fact the initial condition of the DIT is all potentially hazardous commands inhibited, and the test conductors select which commands to "uninhibit.") This allows a flexibility to adjust to various hardware configurations; if an SRB simulator is available, for example, it is not necessary to inhibit the SRB ignition commands.

Even though potentially hazardous signals are prevented from ever leaving the GPC's input/output processor, it is important to verify that such commands would have been issued if the inhibit techniques were not applied. This verification is performed by analysis of telemetry data which reflects the state of the commands prior to application of the inhibit code.

Vehicle instability questions have been raised concerning the execution of DIT with the shuttle attached to fixed supports. It has been suggested that the rate gyros and accelerometers, by the degree of their sensitivity, will detect the movement of the effectors during a DIT. This may cause the vehicle to become unstable if the inputs of these sensors are allowed to propagate through the applications software and drive the effectors. During certain tests, therefore, the substitution mode rather than the combining mode has been elected for these sensors to eliminate this potential hazard.

6 TEST EVALUATION

A DIT is evaluated in much the same manner as a real flight -- by analyzing the standard telemetry, monitoring the cockpit displays, and observing the movement of the vehicle hardware. During a successful DIT, the vehicle should be observed to sequence through the nominal events and GN&C modes associated with the particular flight scenario being simulated.

As would be expected, a set of variables considered sufficient to evaluate a flight is a part of the standard flight telemetry. The closed loop simulated flight (from which simdata is extracted) is also evaluated by analysis of these parameters, extracted from the telemetry recorded during a simulation run. In a like manner, the DIT may be evaluated by comparing similarly obtained data with the corresponding simulation data.

Equally significant verification data comes from the cockpit, so far manned during DIT by Astronauts Young and Crippen as well as other experienced crew members. The cockpit displays provide extensive information to the crew which enables them to monitor the progress of the flight. Of key importance are the caution and warning, and fault annunciation capabilities of the computers, through which the crew receives indication of any subsystem malfunction detected by the GPCs. Any such malfunctions or other discrepancies during DIT are noted for post-test analysis.

The LPS represents another monitoring station where DIT engineers may follow the progress of the test. In particular, the flow of simdata from the LPS to the GPCs is monitored, and any transmission failures are noted. Also, parameters which are not displayed in the cockpit may be examined from the LPS; GPC duty-cycle, for example. Finally, correlation can be made between observed movement of the vehicle aerosurfaces and flight control and aerosurface position feedbacks observed on displays.

7 RESULTS FOR STS-1

The purpose of DIT is to check out the integrated vehicle at KSC. Toward this end, four tests have been run on Columbia and the integrated STS representing the culmination of five years of development work, primarily in the Flight Systems Laboratory (FSL) and the Shuttle Avionics Integration Laboratory (SAIL), mentioned earlier. Perhaps the most basic and singularly most important achievement demonstrated by these tests is that simdata will drive the vehicle. As a result, the technique described here has produced a significant test of the integrity of the shuttle by enabling a dynamic

flight simulation to be performed on the flight-ready vehicle.

The four tests performed at KSC were as follows.

1. Orbiter Integrated Test (OIT), encompassing the period between 16 December 1979 and 18 January 1980, was performed with the Orbiter Columbia horizontal in the Orbiter Processing Facility (OPF). During the course of this test five ascent scenarios were 'flown' as well as a full 48-hour on-orbit operation and a complete re-entry through landing test, all utilizing the redundant primary flight computers. The five ascent runs enabled the test conductors to gauge the effect on Orbiter subsystems of failures to each of the three main power buses, and allowed them to evaluate any differences in performing the tests with ground power versus actual on-board fuel cell power. As with all subsequent tests, the flight crew manned the cockpit and the launch teams were on station in the firing room.

2. Delta OIT, from 7 - 11 July 1980, was essentially an abbreviated rerun of the OIT test after completing the installation of the Orbital Maneuvering and Reaction Control Systems (OMS/RCS). In addition to checking the interfaces to these propulsion systems which had not been available during OIT, it was possible to close out items unresolved by the previous test. One ascent scenario, some selected on-orbit operations, and a descent-to-landing profile were flown.

3. Shuttle Interface Test (SIT), 15-19 December 1980, represented the first full configuration DIT test. It was performed with the Orbiter mated to the External Tank and Solid Rocket Boosters on the Mobile Launch Platform in the Vehicle Assembly Building (VAB). Another milestone reached during this test was the first DIT scenario to be flown utilizing the backup flight computer. As mentioned in Section 1, the crew has the option to engage the backup computer if critical failures are detected in the primary system. Four mission scenarios were exercised. Primary computer ascent and descent tests were conducted as well as two abort modes. Return To Launch Site (RTLS) simulations were performed, one time under control of the primary computers and, in the following test, under backup computer control. An RTLS brings the Orbiter back for an emergency landing at KSC and would be flown, for example, if main engine problems arise soon after launch.

4. Launch Readiness Verification (LRV), 13-16 March 1981, represented the last DIT test prior to STS-1, and was performed with the vehicle on the pad awaiting final countdown to launch. The purpose of this test was to demonstrate the continued integrity of the vehicle after rollout to the launch pad and completion of the Flight Readiness Firing (FRF) of the main engines. LRV encompassed ascent and descent scenarios utilizing the primary computers, and an RTLS mission run using the backup computer.

As a result of these tests, hardware and software problems, both on-board the vehicle and within the launch support complex, were identified and resolved. For example, during SIT one of the main engines shut down prematurely due to a sluggish fuel valve which was subsequently replaced. Significant return has been realized as well in the training of the launch and flight crews. Procedure problems were identified in that certain activities took significantly more or less time to perform than had been estimated. Rewrites of the procedures yielded a more realistic allocation of time and eventually evolved into the STS-1 launch procedure.

Recycle steps were formulated and practiced as certain DIT tests had to be aborted and restarted when problems surfaced at various points in the countdown. Such a procedure was called upon during the first attempt to launch Columbia when a computer problem halted the count at T-9 minutes. In addition, communication and data links connecting KSC with Mission Control in Houston and with the Rockwell support facilities in Downey, California were verified in a flight realistic setting. This proved useful in coordinating problem resolution activities among the various facilities.

In summary, the DIT tests resulted in confidence, not only in the readiness of the Space Shuttle vehicle, but also in the ability of the mission support team to perform as expected and to handle unforeseen circumstances.

8 THE FUTURE OF DIT FOR THE SPACE SHUTTLE

Current schedules call for DIT tests to be performed for the remainder of the shuttle flight tests, STS-2 through STS-4. Discussions are underway to plan for the operational use of the shuttle and to formulate integrated testing needs during this period at both KSC and at Vandenberg Air Force Base in California. In fact, NASA's turn-around time goal of 1-2 weeks from touchdown to next launch may dictate an expanded, near exclusive use of DIT, rather than the traditional subsystem-by-subsystem functional tests.

Computer-automated patch generation techniques have been implemented to greatly reduce the time required for development of the DIT software. This will help to minimize the impact on DIT caused by the introduction of new versions of flight software (FSW). For example, STS-1 used primary computer FSW version 16, STS-2 through STS-4 are planned to use version 18, and a version 19 is anticipated for operational shuttle use starting with STS-5. A change in FSW version usually implies that new requirements for mission performance have been implemented, and also entails a reorganization of the software within the computer memory (i.e., a recompilation and different core mapping). Automation techniques simplify and reduce the labor required to relocate the DIT software when the FSW is upgraded.

An option under consideration as a solution to the same problem is to incorporate the DIT software into the FSW. This could be done by adding logic which will be executed conditional on whether a DIT is to be performed. The DIT software would then be maintained as a part of the FSW (and hence appear in each new FSW version), and not as a separate entity.

Another area under examination which impacts DIT development time is the generation of simdata. As described in Section 3.2, specific closed-loop simulations are performed in which the telemetry content is modified. This is done so that the necessary data is collected to produce simdata, and so that it is available at the proper frequency. Initial studies indicate that it may be possible to synthesize simdata from the existing nominal telemetry by using interpolation and other techniques. This will permit planned software verification tests, independent of DIT, to provide the required input for simdata generation, eliminating the need for special 'modified-telemetry' simulations. Perhaps an even more significant implication of this development would be the ability to replay an actual flight both in the laboratories and on the vehicle. The flight telemetry gathered from STS-1, for example, could be converted into simdata producing a time history of the sensor inputs. It could then be used in the laboratory to step through the flight, allowing close examination of critical flight sequences. This may prove to be an especially useful tool if problems arise during a flight.

Capabilities for DIT are under development or contemplated which will lead to significant operational improvements. In Section 3.4, it was stated that a DIT test must be performed with all clocks (ground and on-board) set to the closed-loop simulation time. If a problem should occur during a test and an unscheduled countdown hold issued, then the test would have to be restarted from the point at which the clocks are set to simulation time. This generally involves a loss of about 4-6 hours. The concept of a variable DIT start (VDS) time has been examined whereby transformations are performed on inertial components of the simdata (i.e., IMU velocities and gimbal angles) to compensate for any given delay in time. In fact, not only will this technique allow for countdown holds beyond the simulation DIT start time, but it makes possible the ability to perform the tests using actual time, hence requiring no clock adjustment whatsoever. The VDS concept is being implemented to support testing for STS-2 and subsequent flights.

Another operational benefit will be gained by the development of an on-orbit DIT capability, currently under study. At present, ascent and descent tests are performed separately, each entailing its own set-up time. An on-orbit DIT capability will permit a smooth and natural transition from ascent to orbit to descent, consolidating the three flight regimes within the framework of one test. The added ability to perform a DIT simulating on-orbit maneuvers as well as payload interface and other activities looks very attractive.

Finally, there is great potential for the extension of DIT to examine off-nominal situations, to perform stress testing and sensitivity analysis. Various failure modes could be simulated to gauge the overall system response, and high noise levels introduced into the simdata sensor inputs to evaluate the resultant perturbations.

9 APPLYING THE DIT/ASIST CONCEPT TO OTHER SYSTEMS

The shuttle is the first completely digitally controlled manned spacecraft, and as such, it introduces a new level of complexity in avionics systems. Throughout the industry, as vehicle complexity and subsystems interactions continue to increase, the need increases to perform fully integrated systems tests which are as close to real flight as possible. The need is compounded by the fact that it has become infeasible in some cases to flight-test a vehicle unmanned, or to perform high-risk or 'throw away' tests.

The ASIST concept, of which DIT is the specific application to the Space Shuttle, presents a promising option to the systems engineer involved in the design and checkout of advanced computer-controlled systems. Given an understanding of the sensory inputs to the system which allow the central computer to perceive its environment, and also an appreciation of system constraints, an ASIST formulation would be possible. As presently conceived, implementation requires a ground (i.e., external) computer with a communication link to the on-board (i.e., system) computer. Not only can ASIST provide a verification of overall system design integrity, but it can also be used to perform routine subsystem maintenance and to help train flight and ground crews. Specific systems which may benefit from an application of the ASIST technique include strategic and tactical aircraft and missiles, other spacecraft and planetary rovers, submarines, and perhaps even air traffic control and nuclear power installations. An extensive discussion of the ASIST concept in its generalized formulation can be found in Reference 1.

DIT for the Space Shuttle was retrofitted to the vehicle, and as such, its development was hampered by the lack of planning for such a test during the early design phase. If an examination of the applicability of ASIST to other systems proves fruitful, it would behoove the designers to plan for such a test. Vehicle or system hardware, software, and support facilities could then be structured to accommodate and facilitate the implementation of ASIST. For example, software should include logic to provide a test path causing alternate, externally supplied sensor inputs (i.e., ASIST simdata) to 'drive' the system. Certainly a safing technique for inhibiting hazardous commands would have to be developed, and could be provided by both software and hardware means. Also, the various support facilities and equipment, including simulators and test sets, should be structured including a requirement to support an ASIST test.

ACKNOWLEDGMENT

The authors gratefully acknowledge the hard work of all the members of the DIT team.

This paper is based upon work performed for Rockwell International Space Division on Purchase Order M8W8XMS-483151R issued under Contract No. NAS9-14000 between Rockwell and the National Aeronautics and Space Administration.

REFERENCES

1. Sripad, A.B. and Ionescu, T.V. "A New Method of Hardware/Software/Mission Integration Testing of Strategic Systems: Case Study For Space Shuttle" submitted for publication to the IEEE Transactions in Aerospace and Electronic Systems.

81-0978

DEVELOPMENT OF A FLIGHT SIMULATION CAPABILITY IN THE DYNAMIC ENVIRONMENT SIMULATOR

WILLIAM B. ALBERY*
AIR FORCE MEDICAL RESEARCH LABORATORY
ACCELERATION EFFECTS BRANCH
WRIGHT-PATTERSON AFB, OH 45433

Abstract

The paper discusses the conversion of the Air Force Medical Research Laboratory's (AFAMRL) human-operable centrifuge from an acceleration stress research device to an operational flight simulator complete with a cockpit and out-the-window visual system. The integration of a Digital Avionics Information System (DAIS) with the facility and the unique problem of transmitting signals over three sets of slip rings is discussed. Using existing hardware and software at Air Force Wright Aeronautical Laboratories, AFAMRL is turning their centrifuge into a flight simulator.

Background

The Dynamic Environment Simulator (DES) was developed for the Air Force Aerospace Research Laboratory at Wright-Patterson AFB during the 1960s. It is a high acceleration (up to 20 G's) centrifuge originally designed and used to do a variety of research programs on physiological acceleration stress. Both human and animal subjects have been used in these studies. As the mission objectives of the AFAMRL have changed to require more operationally oriented efforts, the capabilities of the DES as an effective support facility have diminished. Over the past fifteen years, advances in aircraft technology have resulted in aircraft designs that have pushed the aircrew into stress regimes never before experienced. Direct lift, direct side force, control configured vehicles, higher thrust-to-weight ratios due to lighter materials and more powerful engines have all combined to expand the performance envelope of the fighter and attack aircraft. As a result, a need has evolved to address many of the design, development, test, evaluation, tactical, procedural and safety issues resulting from this evolution in an operationally realistic, but safer environment. The objective of this update is to make the DES more operationally realistic.

Dynamic Environment Simulator (DES)

The DES (Fig 1) has three axes of movement. The main arm (20 ft. long) turns around one axis, the fork rotates on a second axis and the cab, or gondola (10 ft. in diameter), rotates on a third axis. The DES generates a horizontal G by the turning of the arm. This G is related to the angular velocity (ω) of the arm by the following:

$$G = r\,\omega^2$$

where G is the angular acceleration (ft/sec^2), r is the length of the arm and ω is the angular velocity of the arm in radians per second.

The horizontal G resulting from the arm movement can be combined with the 1 G from normal earth gravity to produce a total G that the subject will actually feel in the cab.

By rotating the cab and rotating the fork (or equivalently, tilting the seat), this total G vector can be broken up into three components (G_x, G_y, G_z) with respect to the subject. Theoretically, with the proper cab/fork/seat orientation, high G forces can be produced in any of the three axes.

In addition, the DES can produce a tangential acceleration, simply by accelerating the arm. If the subject is sitting upright in the cab (i.e., no tilt in the seat), he would experience this linear acceleration as a G_x. However, this tangential G would never exceed 40% of the acceleration of gravity.

The aforementioned G components (G_x, G_y, G_z) can be read from accelerometers on the DES system.

There are, of course, limits on the dynamics of each of these movements. The arm has a maximum speed of about 55 revolutions per minute (RPM). This is enough velocity to produce over 20 G's. There is a man-rated maximum of 9 1/2 G's. The maximum angular acceleration (α) of the arm varies with its speed. Zero to six radians per second includes a range of G forces from zero to over 20. To avoid the complications arising from such variations, a set limit of 0.75 G per second is presently being adhered to by the DES personnel. Even so, this 0.75 G per second rate can only be sustained for about 20 seconds without overheating the arm motors. As expected, the limits on the capability of the DES to rapidly change G's is the single most significant limiting factor in simulating a high performance aircraft (e.g., an F-16) which can change G's very quickly.

The fork is rarely used and consequently, there is little data available on it. It has a maximum speed of 30 RPM but its inertia is so large and its motor so small (90 hp) that its acceleration is generally considered to be minimal. The cab has a maximum speed of 150 RPM but its maximum safe speed is 30 RPM. Its maximum acceleration is 6.3 rad/sec^2 which is very high compared to the arm and the fork.

THE PRESENT DES SYSTEM

A simplified block diagram of the current DES closed loop system is shown in Figure 2. There are three computers: a hydrid (EAI 3200 analog and a Pacer 600 digital), a PDP 11/34, and a PDP 11/40. In simple terms, the hybrid computer does the simulation, pilot scoring,

*Electronic Engineer; DES Update Manager
Member AIAA

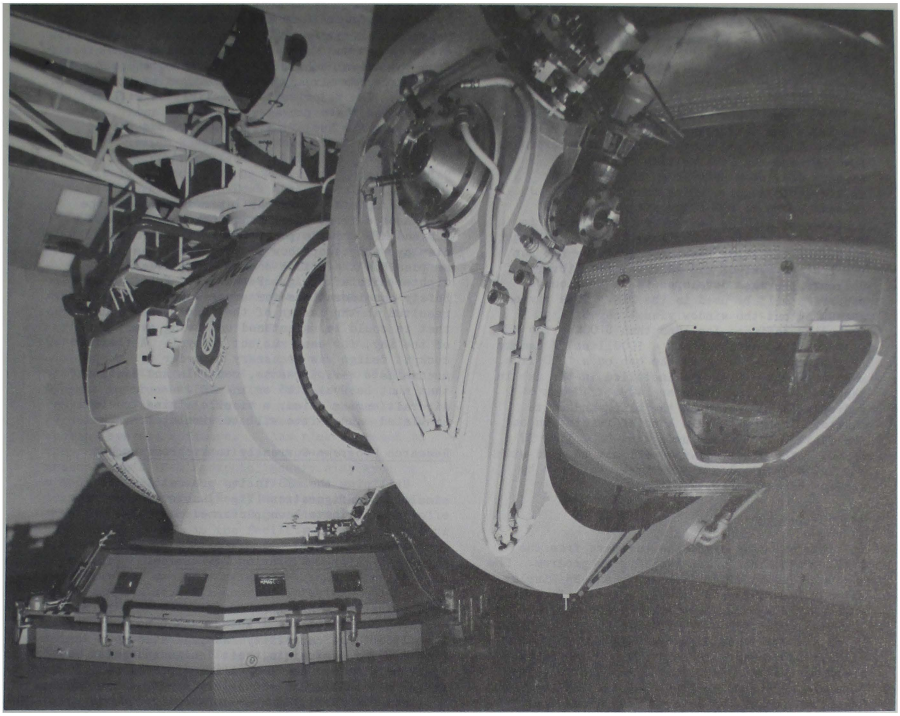


FIG 1 - THE DYNAMIC ENVIRONMENT SIMULATOR (DES)

error tracking, and forcing function creation. The PDP 11/34 drives the graphics display which by means of a camera is transmitted into the cab. Finally, the PDP 11/40 performs a number of functions, one of which is to provide the interface between the simulation and the DES itself. It accepts the simulation signal, for a particular roll, pitch, or G force from the hybrid and tailors the signals to ones which the DES can accept.

The graphics system can produce a target which can have one of two degrees of freedom, either vertical or horizontal. The control stick in the cab allows the subject to track the target in either of two modes, longitudinally or laterally, which in turn produce the desired G forces on the subject.

In the open loop system, the subject having no control, a G profile can be generated, and the DES will follow this predetermined run exerting the required G forces.

Turning the DES into a Flight Simulator

To achieve the basic goals of the update there are three main areas of the DES facility that need upgrading. These areas are (1) graphics generation, (2) the hybrid (or simulation) computer and (3) the cab interior. The graphics generation upgrade is being accomplished by the addition of an out-the window visual system and by integrating an operational avionics system with the computer and cab; the hybrid computer upgrade was accomplished by the addition of a SEL 32/77 to replace the Pacer 600 and perform the processing of avionics and flight program software; the cab interior is being upgraded by the addition of a partially-operational instrument panel with a Heads-Up Display and an advanced aircraft seat.

Throttle, stick and rudder pedals will be operational. The elements of this update is depicted in Figure 3.

Visual System

The visual system selected for the DES is the Evans and Sutherland Multi-Picture System (MPS). The MPS was chosen for two basic reasons: low cost and the availability of already-developed software. The MPS is a calligraphic display which is capable of generating up to 400 lines every frame. It operates at a 60 Hz refresh rate. The MPS is used for a variety of applications and at Wright Patterson Air Force Base, the Avionics Lab has two Picture Systems and an extensive library of software packages developed for the Avionics Lab/WPAFB will be used in the DES. This gives AFAMRL an out the window visual capability including software for less than $100K. The MPS video will be picked up by a 1000 line TV camera (Fig 3) and displayed in the cab on a 1000 line monitor until optics are installed in the cab in the future. Since there are 4096 x 4096 x 64 addressable locations on the black and white monitor, there will be a loss of resolution on the monitor in the cab using the 1000 line TV system. The visual system will not be used with the Heads-Up Display since the TV will not be viewed through infinity optics.

Avionics System

The avionics system is hardware from the Digital Avionics Information System program. The hardware includes two displays (Head-Up Display and a Multi-Purpose Display), a Remote Terminal, a Modular Programmable Display Generator, a Display Switch/Memory Unit and a DAIS processor. After the DAIS equipment was selected for the DES, it was decided to load most of the hardware on-board the DES and to establish the interface between the Remote Terminal and the Bus Control Unit over one of the twisted pair lines that travel from the computer room, through the three slip rings and into the cab (Fig 3). This decision was made in order to reduce the requirements on available video lines (there are only 4). However, in establishing this interface, the requirement to transmit MIL-STD-1553B quality signals over the DES' twisted pair lines had to be tested. 1553B bus tests were performed over the twisted pair with the cab static and dynamic. Based upon the results of these tests, it was concluded that a 1553B data bus can be used reliably for processor-to-processor communication in the DES.

Computer Hardware and Software

The principal computational components of the computer upgrade of the DES include the PDP 11/40 which will drive the Multi-Picture System, the SEL 32 which will drive the avionics system and process the flight software, the EAI 3200 which performs hybrid activities and the interfaces between the SEL 32 and the avionics system (Fig 3). Software added to the DES includes a DEC RSX-11M Operating System, F-16 Dynamics Software from the Air Force Flight Dynamics Laboratory, DAIS software from the Avionics Lab and Multi-Picture System software from the Avionics Lab. Most of the software for the update has already been developed by outside sources. The major software activities will be in adapting these packages to the DES and developing the Real-Time Processing load which will operate the system.

Cab Interior

The update of the cab interior includes the design of a front panel framework that supports a HUD and MPD and the positioning of controls, displays and canopy bow to approximate as closely as possible the actual cockpit of a modern fighter aircraft (e.g., F-16, F-18, F-14). Safety considerations for emergency egress resulted in the design of the front panel such that it could be unfastened and swing over, out of the way, for easy subject removal. A flexible cockpit design was necessary in order to accommodate various seats, controls and equipment that must be evaluated on the DES. Aircraft/engine noise, a missile aiming tone and simulated cannon fire will be simulated.

Research Programs Currently in Progress on DES

Utilizing the DES in its present limited simulator configuration (Fig 2) some of the efforts which have been performed recently include:

AFTI/F-16 Acceleration Performance Evaluation

This effort is investigating the effect on pilot performance with lateral acceleration up to ±2Gy. This side force capability has generated new problems for restraints, seat design and pilot tracking. Pilots use the existing rudder pedals in the DES to track a simulated target on the cab monitor.

AFTI/F-16 Voice Controlled Input Device System (VCIDS)

The VCIDS effort is being conducted at acceleration levels up to +6Gz and in combined environments up to +4Gz combined with ±2Gy. The present visual system is not used in this experiment. Voice commands under acceleration stress are being evaluated for incorporation of the VCIDS into the F-16.

KFIR-C2 Flat Spin Indoctrination

Simultaneous +Gz and -Gx accelerations were involved in this effort requiring the full digital control capabilities of the DES. An essentially complete cockpit mockup was installed including throttle, center stick and instrument panel. The spin environment was found much more stressful than Israeli pilots expected it to be.

Future DES Programs

When the update of the DES is complete,

later this year, the scope of research topics will be broadened greatly by the addition of visual and avionics systems. Some of the additional projects that will be performed once the flight simulation capability is available in the DES:

- Air to Air Combat
  - Medium Range Missile Attack
  - Short Range Missile Attack
- Air to Surface Weapons Delivery
  - Level or Dive-Toss Attack
  - Pop-Up Attack
- Threat Evasion, SAM Break
- Terrain Following/Terrain Avoidance

The HUD and out-the-window visual system will make these simulations possible. Previous experiments used the very simple HUD simulation capability of the PDP 11.

Conclusion

A relatively low cost update to the AFAMRL Dynamic Environment Simulator ($300K) has resulted in the addition of a visual system, avionics system, cockpit enclosure and flight and control systems software. A low risk approach to the out-the-window visual system was accomplished. Calligraphic imagery displayed on a monitor is picked up by a TV camera and transmitted over an already existing coaxial line into the cab of the centrifuge. Avionics equipment, donated to the facility is mounted on-board the cab and driven remotely over twisted pair carrying MIL-STD-1553B quality data. The 10 foot diameter cab accommodates various aircraft seats and can accept a cockpit structure and physiologic equipment. An advanced fighter front panel has been designed for the updated cab. Flight and control systems software from the Air Force Flight Dynamics Laboratory's F-16 simulation was utilized because of compatibility and no cost. The addition of a SEL 32/77 computer to handle avionics and flight software was also accomplished. The evolution of the DES from a physiological stress device to an acceleration stress flight simulator will be complete in late 1981.

References

Hoffman, M., Honaker, J., Nelson, M., Woodruff, K. Analysis for Upgrade of the Dynamic Environment Simulator (DES), Feb 80, Systems Research Laboratories, Dayton, OH 45440.

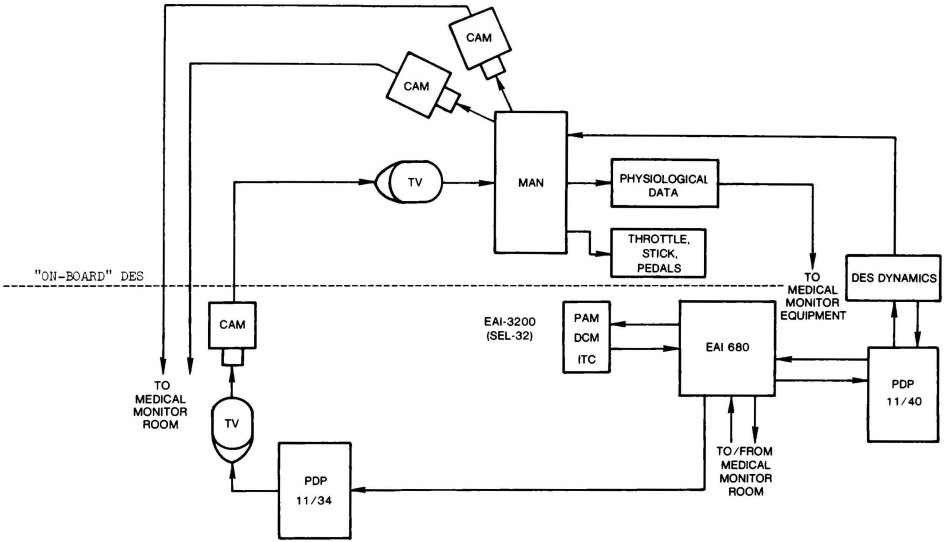


FIG 2 - PRESENT DES SYSTEM CONFIGURATION

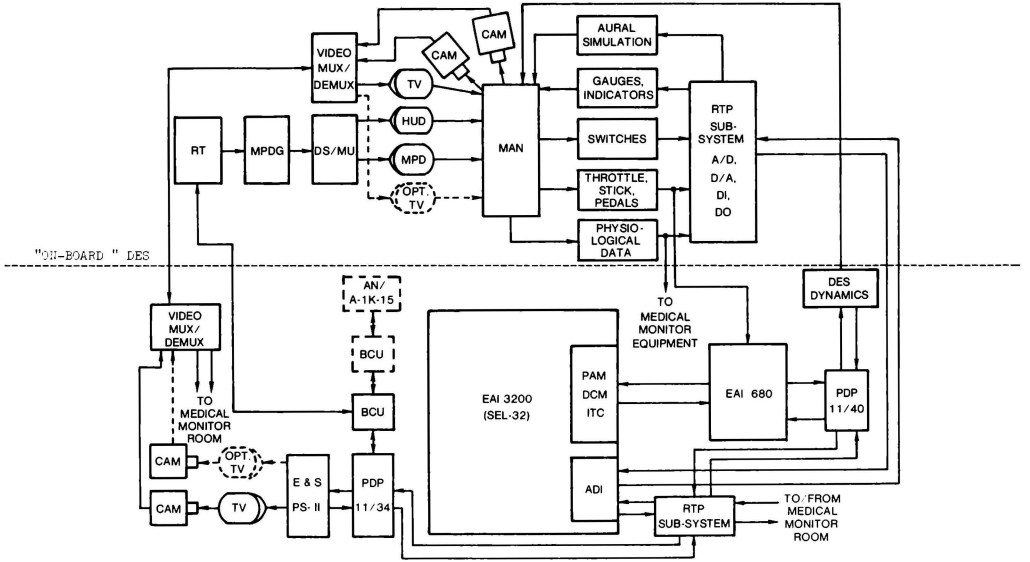


FIG - UPDATED DES SYSTEM CONFIGURATION

FORMAT FOR MATHEMATICAL MODELS USED IN COMPUTER SIMULATION

George A. Zetkov*
Grumman Aerospace Corporation
Bethpage, NY

Abstract

A format for mathematical models used in computer simulation is presented. This format is applicable to a complete system or any subsystem within. The methods considered for specifying processing (i.e., mathematical operations) include the use of 1) flowcharts and 2) tables. The parameters considered for specifying each data item are: 1) name, 2) word description, 3) arguments, 4) value, 5) iteration rate, etc. An example of the use of the recommended format is provided.

Nomenclature

c	Cover for optical sight
d	Designated platform number
E	Engage mode at system
f_{PPI}	Flag: PPI on (0 = no, 1 = yes)
$f_{PPI}(p)$	Flag: PPI shows platform p (0 = no, 1 = yes)
i	Iterations per second
m	Mode of system (1 = ready, 2 = engage)
p	Platform number
PPI	Plan position indicator
r(p)	Slant range of platform number p relative to ownplatform
$r_{PPI}(p)$	PPI range of platform number p
R	Ready mode of system
s_m	Switch for system mode (1 = ready, 2 = engage)
T( )	At transition into the condition described within the parentheses
t_E	Time elapsed in the engage mode, seconds
x(p)	Distance east of GAO (gaming area origin), platform number p

I. Introduction

When one group of people performs the function of formulating mathematical models for computer simulation and then presents these models to another group of people that performs the function of programming and coding these mathematical models, communication problems can arise. Examples of such problems are as follows:

- The mathematical model is inadequately expressed in the sense that equations are given for some of the modes of operation but are omitted for other modes of operation.
- The characteristics of data items are inadequately defined.
- Data item characteristics as given in one part of a model contradict data item characteristics as given in another part of the model.
- The mathematical model is too detailed in the sense that in addition to the appropriate equations, details are specified which unnecessarily restrict the basic approach, structure, or sequence of the programming in some way.

The communication problems can be minimized by constructing the mathematical model in a suitable format, a format in which there is a prompting of all the different types of details required and in which there is a standardized structure which enables a programmer to convert the information rapidly and efficiently into computer code. A format considered suitable for the expression of mathematical models used in computer simulation is presented. This format is applicable to a complete system or any subsystem within a complete system.

II. Definition of the Processing

The definition of the mathematical model is divided into two parts: 1) definition of the processing and 2) definition of the data (variables and parameters). The definition of the processing is discussed in this section, whereas the definition of the data is discussed in the next section. The processing consists of the computations that are performed to obtain the present state of the intermediate and output variables in the system. The two methods considered for the definition of the processing are the use of flowcharts and tables.

Processing Type 1

The first type of processing examined is the type in which a set of computations is activated once by an event. This type of processing is shown in Fig. 1. As shown in this figure, when event 1 occurs, then computation set 1 is performed once. When event 2 occurs, then computation set 2 is performed once. This cycle is then repeated.

The flowchart format for this type of processing is shown in the example of Fig. 2. In the flowchart format,there are two kinds of operation: one type indicated by the line enclosures that are shaped in the form of a baseball diamond, ◇ , and the other type indicated by the line enclosures that are shaped in the form of a rectangle, □ . The diamond indicates that one of a number of computation paths is to be taken, depending on the values of particular variables. The rectangle indicates that intermediate or output variables are to be computed. By the arrangement of the diamonds, rectangles, the connecting lines between them, and by the information contained within the diamonds and rectangles, the flowchart shows the sequence of computations to be performed.

The symbols c, m, and t_E, which appear in Fig. 2, are defined in the preceding section, Nomenclature, but only in a general sense. The detailed characteristics of each variable and parameter in a model must be defined as part of the mathematical model; however, this subject is deferred until the next section, III. Definition of the Data.

---

*Engineering Specialist

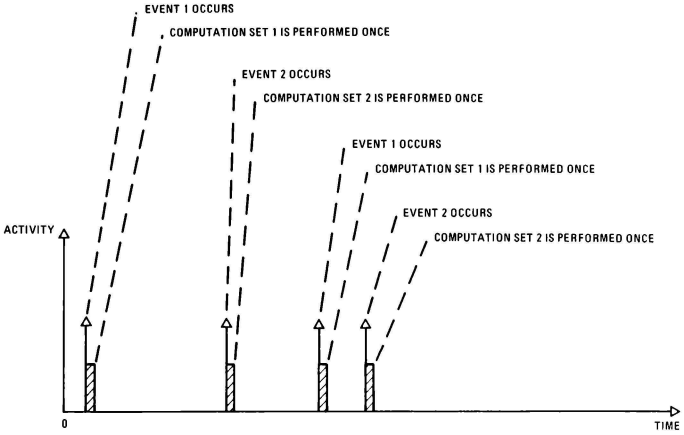


Fig. 1 Processing Type 1

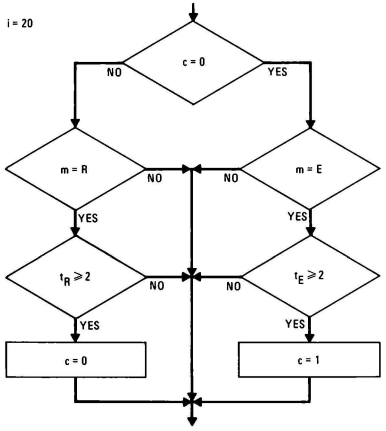


Fig. 2 Example of Processing Type 1 Using the Flowchart Format

As indicated at the top of the flowchart, the flowchart computations are to be performed 20 times each second. The flowchart indicates that the following operations are to be performed. If the cover for the optical sight is closed (c = 0) and if the system mode is in engage (m = E) and if the time elapsed in the engaged mode is equal to or more than 2 seconds ($t_E \geq 2$), then the cover is changed from closed to open (c is set to 1). Returning to the top of the flowchart, if the cover is open ($c \neq 0$, i.e., c = 1) and if the system mode is in ready (m = R) and if the time elapsed in the ready mode is equal to or more than 2 seconds ($t_R \geq 2$), then the cover is changed from open to closed (c is set to 0). For any case other than the two cases just described, the cover is left in its existing state, whether it is closed or open.

The same example that is defined in the flowchart format of Fig. 2 is also defined in the table format, as shown in Fig. 3. Referring to this table, within 0.1 second after the event, at two seconds after transition into the engage mode, the cover is set at open (c = 1). Similarly, within 0.1 second after the event, at two seconds after transition from the engage mode to the ready mode, the cover is set at closed (c = 0).

EVENT	COMPUTATIONS PERFORMED WHEN EVENT OCCURS	MAXIMUM DIFFERENTIAL BETWEEN EVENT AND COMPUTATIONS (SEC)
AT TRANSITION INTO MODE ENGAGE PLUS 2 SECONDS	c = 1 (COVER OPEN)	0.1
AT TRANSITION FROM MODE ENGAGE INTO MODE READY PLUS 2 SECONDS	c = 0 (COVER CLOSED)	0.1

Fig. 3 Example of Processing Type 1 Using the Table Format

Comparing the table of Fig. 3 with the flowchart of Fig. 2, the event column of the table is equivalent to the diamonds and the lines leading into and out of the diamonds of the flowchart and the computation column of the table is equivalent to the rectangles of the flowchart. If, the model formulator (one who generates the model) uses the flowchart format, the model formulator must devise an arrangement of diamonds, rectangles, connecting

lines, and information within the diamonds and rectangles, as well as an indication of the iteration rate at which the flowchart operations are to take place, to define the model. The resulting arrangement may or may not represent the most efficient method of determining whether or not an event has occurred. On the other hand, if the model formulator uses the table format, the sequence of computations to determine whether or not an event has occurred is not required; instead, this sequence is left up to the programmer. Thus, the table format has the advantage of not requiring the generation of any specific sequence of computations to determine whether or not an event has occurred. Also, the use of the table format has the advantage of not requiring the drawing of the diamonds, rectangles, and connecting lines that are used in the flowchart format. Forms can be made for tables, but not for flowcharts.

Processing Type 2

The second type of processing examined is the type in which the performance of a set of computations is activated when a specified event occurs, is then repeated periodically at a specified iteration rate, and is terminated when a specified event occurs. This type of processing is shown in Fig. 4. As shown in this figure, when event 1 occurs, then computation set 1 is performed immediately and repeated periodically at the prescribed iteration rate until event 2 occurs, at which time computation set 1 is no longer performed. The cycle is then repeated.

The flowchart format for this type of processing is shown in the example of Fig. 5. As indicated at the top of the flowchart, the flowchart computations are to be performed once per second. The flowchart indicates that the following operations are to be performed.

The flag f_{PPI} indicating the status (off or on) of the PPI (plan position indicator) is checked first. If the PPI is on (operating) and if the slant range r(p) of platform number p relative to ownplatform is within PPI detectable range limits ($0.5 \text{ km} \leq r(p) \leq 20.5 \text{ km}$), then the flag $f_{PPI}(p)$ is set to the value of one, indicating that platform p is within PPI detectable range, and the slant range r(p) for platform number p is transformed into the PPI range $r_{PPI}(p)$ for platform number p. (Under the definition of the characteristics of variables and parameters, the range of values for r(p) is 0 to 100 km, with a resolution of 0.1 km, whereas the range of values for $r_{PPI}(p)$ is 1 to 20 km, with a resolution of 0.5 km.) If a platform is not within PPI detectable range limits, then the flag $f_{PPI}(p)$ is set to the value zero (for use by firmware driving a simulated PPI). Referring to the top of the flowchart again, if the PPI is not operating ($f_{PPI} = 0$), no additional computations are required.

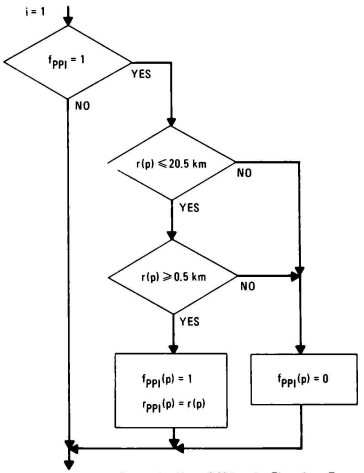


Fig. 5 Example of Processing Type 2 Using the Flowchart Format

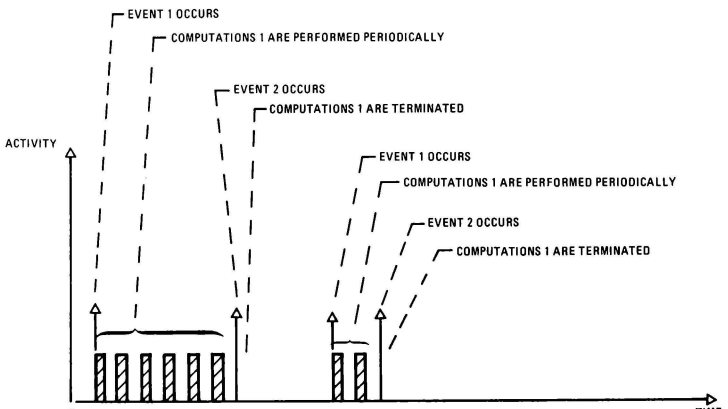


Fig. 4 Processing Type 2

The same example that is defined in the flowchart format of Fig. 5 is also defined in a table format, as shown in Fig. 6. As indicated in the table, when the PPI is turned on ($T(f_{PPI} = 1)$), the computations shown in column 2 are activated, being performed repetitively at the iteration rate of once per second as specified in column 4. When the PPI is turned off ($T(f_{PPI} = 0)$), the computations shown in the first row of column 2 are no longer performed.

Comparing the table of Fig. 6 with the flowchart of Fig. 5 is also defined in a table for- the same as those described for processing type 1.

1	2	3	4
EVENT	COMPUTATIONS ACTIVATED WHEN EVENT OCCURS AND REPEATED AT RATE OF COLUMN 4	MAXIMUM TIME DIFFERENTIAL BETWEEN EVENT & COMPUTATIONS (SEC)	ITERATIONS PER SECOND FOR COMPUTATIONS OF COLUMN 2
$T(f_{PPI} = 1)$	IF $0.5 \leq r(p) \leq 20.5$ km: $f_{PPI}(p) = 1$ $r_{PPI}(p) = r(p)$ ELSE: $f_{PPI}(p) = 0$	0.1	1
$T(f_{PPI} = 0)$		1	

Fig. 6 Example of Processing Type 2 Using the Table Format

III. Definition of the Data

In addition to defining the processing, the characteristics of each variable and parameter are defined. A variable is categorized as either an input, intermediate, or output variable. An input variable is a variable that is input into the model being defined; an output variable is a variable that is output from the model being defined; and an intermediate variable is any other variable in the model being defined.

The characteristics of each variable and parameter are defined by inserting appropriate entries into a table of the form shown in Fig. 7. Referring to this figure, column 1 refers to the mathematical model being defined, relative to which a variable is input, intermediate, or output. Referring to columns 2 through 5, a check is inserted into one or more of the columns, indicating whether the item is a variable or parameter and if the item is a variable, whether the item is an input, intermediate, or output variable. In some cases, a variable is both an input and an output variable (e.g., input because its previous value is used to compute its present value and output because its present value, when computed, is put into a file for use in the next iteration). In column 6, the mathematical symbol used to represent the variable or parameter is entered. Alternatively, if the model formulator uses an acronym instead of a mathematical symbol, column 6 is left blank and column 7 is entered. If the model formulator has entered a mathematical symbol in column 6, the insertion of an acronym in column 7 is optional; i.e., the model formulator may insert his preference for a portion of the software name (e.g., up to four alphanumerics are inserted. The programmer creates a software mnemonic by using the acronym inserted and adding two alphanumerics to the front of the acronym). In column 8, the item is defined in words. In column 9, the arguments of a variable (those independent variables of which the variable is a function) are entered. (Note: The characteristics of each argument are also defined.) If the item has a numerical value, the units (dimensions) of the item are entered in column 10, its range of values is entered in columns 11 and 12, its resolution is entered in column 13, and its required accuracy is entered in column 14. If the item has a logic state, the symbol for each logic state is entered in column 15 and the definition of each symbol is entered in column 16. In column 17, the iteration rate for each intermediate or output variable (processing type 2) is entered. If there is more than one iteration rate, each iteration rate is inserted versus its associated conditions. In column 18, the source of an input variable is a mathematical model or hardware (system, panel, component, element) which outputs this variable. (A column heading for the destinations of a variable output is not included, because the output is listed as an input in other mathematical models or in hardware being defined.)

With regard to the example data inserted in Fig. 7, the mathematical model, Electro-Optical System, is being defined. The first item is the input m, the system mode, which is in one of the two logic states ready and engage, and which is generated in the model, System Mode. The second item in the table is the input t_E, the time elapsed in the engage mode, which has a range of values from 0 to 30,000 seconds, with a resolution of 0.1 second, and is generated in the model, System Mode. The third item in the table is the output c, the cover for the optical sight, which is either closed or open.

A second example of data definition is given in Fig. 8. In this case, the mathematical model, Platform Motion, is being defined. The first item is an intermediate variable, represented by the symbol p, denoting the platform number. The variable can take on any value from 1 to 10 in increments of one, being initialized to the value 1 within the model and indexed up one at a time until the value 10 is reached. The second item in the figure is a variable that is both an input and an output (an input because the value of the variable at the previous iteration is an input to the present computation of the variable and an output because the present computation of the variable is put into a file for use as an input in the next iteration). The symbol for the variable is x(p), which denotes the distance east relative to the gaming area origin of platform number p. Thus, the variable is a function of the argument p. The iteration rate is once per second for condition a (the platform number p is not equal to the designated platform number d) and is thirty per second for condition b (the platform number p is equal to the designated platform number d).

A third example of data definition is given in Fig. 9. In this case, the mathematical model, PPI, is being defined. The first item is an input variable, r(p), denoting the slant range of platform number p relative to the ownplatform. The variable can have a value from 0 to 100 km in increments of 0.1 km and is updated once every second. The second item in the figure is an output variable $r_{PPI}(p)$, denoting the range of platform number p as shown on the Plot Position Indicator. The argument at the variable is therefore the platform number p. The variable can have a value from 1 to 20 km in increments of 0.5 km and is updated once every second.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
MATH MODEL	VARIABLE			PARAMETER	MATH SYMBOL	ACRONYM	DEFINITION	ARGUMENTS	NUMERICAL VALUE					LOGIC STATE		ITERATIONS PER SECOND (PROCESSING) TYPE 2)	SOURCE OF INPUT	NOTE
	INPUT	INTERMED	OUTPUT						UNITS	RANGE FROM	RANGE TO	RESOLUTION	ACCURACY	SYMBOL	DEFINITION			
ELECTRO-OPTICAL SYSTEM	√				m		MODE OF SYSTEM							1 2	READY ENGAGE		MODEL: SYSTEM MODE	
	√				t_E		TIME, ENGAGE MODE		sec	0	30,000	0.1					MODEL: SYSTEM MODE	
			√		c		COVER, OPTICAL SIGHT							0 1	CLOSED OPEN			

Fig. 7 Characteristics of the Variables and Parameters

MATH MODEL	VARIABLE			PARAMETER	MATH SYMBOL	ACRONYM	DEFINITION	ARGUMENTS	NUMERICAL VALUE					LOGIC STATE		ITERATIONS PER SECOND	SOURCE OF INPUT VARIABLE	NOTE
	INPUT	INTERMED	OUTPUT						UNITS	RANGE FROM	RANGE TO	RESOLUTION	ACCURACY	SYMBOL	DEFINITION			
PLATFORM MOTION		√			p	p	PLATFORM NUMBER			1	10	1						AN ARGUMENT INDEXED WITHIN THE MODEL.
	√		√		x (p)		DISTANCE EAST OF GAO	p	nm	0	20	10^{-4}	10^{-2}			1 IF (a), 30 IF (b)		(a) p ≠ d, (b) p = d

Fig. 8 Characteristics of the Variables and Parameters

MATH MODEL	VARIABLE			PARAMETER	MATH SYMBOL	ACRONYM	DEFINITION	ARGUMENTS	NUMERICAL VALUE					LOGIC STATE		ITERATIONS PER SECOND	SOURCE OF INPUT VARIABLE	NOTE
	INPUT	INTERMED	OUTPUT						UNITS	RANGE FROM	RANGE TO	RESOLUTION	ACCURACY	SYMBOL	DEFINITION			
PPI	√				f_{PPI}		FLAG: PPI ON							0 1	OFF ON		HARDWARE: PPI	
	√				r(p)		SLANT RANGE	p	km	0	100	0.1					MODEL: PLATFORM MOTION	
			√		$r_{PPI}(p)$		PPI RANGE	p	km	1	20	0.5				1		SCALE = 2/km. UNSIGNED BINARY INTEGER 2 TO 40.
			√		$f_{PPI}(p)$		FLAG: PLATFORM ON PPI	p						0 1	OFF ON	1		

Fig. 9 Characteristics of the Variables and Parameters

As required by the PPI simulator, the scale for rPPI(p) is 2 per kilometer, with the final result as an unsigned binary integer with a value from 2 to 40.

IV. Definition of the Test Plan

To verify that the computer program corresponding to the mathematical model actually performs the operations and generates the data defined in the mathematical model, verification test procedures are synthesized by the model formulator. These procedures may be defined by using a format similar to that shown in the example of Fig. 10.

TIME		INPUT						OUTPUT					NOTE
MIN	SEC	s_m	m					m	t_E SEC	c			
0	0	R	R										
1	0							R		0			
2	0	E						E	0.	0			
2	2							E	2.	1			

Fig. 10 Vertification Test Plan

As shown in this figure, at the time of zero, the inputs inserted are: the mode switch is set at ready (s_m = R) and the mode is set at ready (m = R). At the time of one minute, the outputs to be observed are: the mode is in ready (m = R) and the cover for the optical sight is closed (c = 0). At the time of two minutes, the input to be inserted is: the mode switch is set at engage (s_m = E). The outputs to be observed are: the mode is in engage (m = E), the time elapsed in the engage mode is zero (t_E = 0.), and the cover for the optical sight is closed (c = 0). At the time of two minutes plus two seconds, the outputs to be observed are: the mode is in engage (m = E), the time elapsed in the engage mode is two seconds (t_E = 2.), and the cover for the optical sight is open (c = 1).

V. Conclusions

Formats for defining a mathematical model have been presented. The model is defined in a manner to permit the transformation of the model into a digital computer program. The model includes: 1) the formula for each output and intermediate variable 2) the characteristics of each input, intermediate, and output variable and each parameter. In addition to the definition of the mathematical model, a format is provided for a verification test plan to demonstrate the capability of the computer program to perform the functions of the mathematical model.

The commonality among the three formats in the above items is the use of a table (see Figs. 3, 6, 7, and 10). Column headings are fixed in all cases, except for the verification test plan in which some of the column headings are entries made by the model formulator. Essentially, the entries made by the model formulator are rows of information under the column headings.

VI. Acknowledgment

This paper has been written in keeping with the major principles put forth in Ref. (1).

VII. Reference

1. K. L. Heninger, "Specifying Software Requirements for Complex Systems: New Techniques and Their Application", IEEE Transactions on Software Engineering, Vol. SE-6, No. 1, January 1980.

DEVELOPMENT OF A GENERIC AIRPLANE RESPONSE SIMULATION

Charles F. Suchomel and David J. Moorhouse

AF Wright Aeronautical Laboratories
Wright-Patterson AFB, Ohio 45433

ABSTRACT

Piloted simulation is a major tool for accomplishing flying qualities research. Two aspects constitute a significant problem: 1) the flying qualities specification criteria are written in terms of classical response modes - we have the continuing task of improving the criteria and their applicability to future airplane developments. 2) a simulation of an actual system (F16, F-18, etc.) is not a good research tool because it is extremely difficult to vary the response parameters in a controlled manner (as opposed to varying aerodynamic or FCS parameter with an indirect effect on dynamic responses). If we also consider independent 6 degree-of-freedom control (the addition of direct lift control and direct sideforce control) the problem is even more severe. Our solution for flying qualities research is an airplane simulation model which is formulated directly in terms of response variables. This is achieved by combining linear transfer function representations of response to control and disturbance inputs with the nonlinear inertial terms. The development and formulation of the simulation model is presented in detail. The rationale, assumptions and limitations are discussed. An apparent limitation, indicated by the use of linear transfer functions, can be removed by scheduling parameters of the transfer functions with other response variables to achieve a level of complexity appropriate to the experiment being performed. This capability to tailor the model complexity to the needs of the individual experiment is discussed. We see uses of the simulation model beyond flying qualities research: as a direct aid in control law development, as a means to investigate the interaction between airplane dynamics and the display, as a simplified aircraft model in combat simulations, etc. A discussion of the validation and potential uses of the model concludes the paper.

NOMENCLATURE

a_{ij}	Elements of aircraft response to controller transfer function matrix: i=1, ---, 6 and j=1,---, 6
e_{ij}	Constants to control magnitude of velocity coupling allowed in simulation; values of $0 \leq f \leq 1$, i=1, 2 3
f_{ii}	Constants to control magnitude of gravity effects allowed in simulation; $0 \leq f \leq 1$, i=1, 2, 3
g	Acceleration due to gravity
I_x, I_y, I_z	Moments of inertia about the x, y, z body axes
I_{xy}, I_{xz}, I_{yz}	Products of inertia
m	Vehicle mass
L, M, N	Roll, pitch and yaw moments acting about the x, y and z body axes respectively
P, Q, R	Roll, pitch and yaw rates about the x, y and z body axes respectively
p, q, r	rotational velocity perturbations about the x, y, z body axes respectively
s	Laplace operator
t	Time
u, v, w	Velocity perturbations along x, y, and z body axes, respectively
U, V, W	Longitudinal, lateral and vertical velocities parallel to x, y and z body axes
X, Y, Z	Forces acting in direction of x, y and z body axes, respectively
Ψ, Θ, Φ	Yaw, pitch and roll Euler rotation angles vehicle body axes to inertially fixed axes
λ	General perturbed state variable used to represent any one or all of the variables u, v, w, p, q, r
δ	General perturbed control variable
T	General symbols used to represent any one or all of the forces or moments acting on the aircraft
d( )	Signifies total perturbation
x, y, z	Body axes forming a right handed Cartesian system.

Subscripts

$(\dot{})$	Time rate of change
o	Initial condition or operating point
g	Atmosphere perturbation due to gusts, wind shear or wind turbulence

I. Introduction

The art or science of Flying Qualities involves consideration of pilot interactions with the airplane stability and control characteristics, flight control system functions, the displays, the cockpit control manipulators, the atmospheric environmental conditions and (almost) anything else that affects the way in which the pilot accomplishes the mission or task. The stability and control characteristics, both natural and augmented, govern the airplane response characteristics that the pilot is attempting to control. The flight control system includes augmentation, pilot assist and fully automatic functions. Normal operation must be satisfactory, and failure-mode operation acceptable to the pilot or at least safe. The displays must present useful information to the pilot in a form that can be assimilated. Pilot workload and mission performance can be affected beneficially or adversely by the displays. The control manipulators are required to give the correct force and deflection cues to aid the pilot in maneuvering the airplane. Requirements on all these factors, the interactions and their effects on pilot workload and mission performance, i.e. the flying qualities, form design criteria for a large portion of the airplane system. The flying qualities criteria are documented in a military specification[1]; more insight into its development and use can be obtained from Ref 2.

A large data base is necessary for confident definition of the many parameters which determine whether the flying qualities of a particular combination are satisfactory, acceptable, or unacceptable. Accumulating and correlating this data base in terms of general criteria gives a designer: (i) the range of values of a particular parameter, or combination of parameters, that yields satisfactory (Level 1) flying qualities; (ii) values that yield flying qualities worse that Level 1 but appropriate for certain failure states or off-design flight conditions; (iii) the sensitivity of flying qualities to variation in particular parameters i.e., how well the optimum is defined or how flying qualities degrade for parameter variations away from the optimum; (iv) the criteria to tradeoff rationally the various factors influencing the flying qualities.

Conventional aircraft give the pilot the control capability to produce angular accelerations about all three body axes, plus longitudinal acceleration; i.e., the pilot only has independent control of four of the six basic airplane states - vertical and lateral translation are dependent functions of airplane angular states. There are potential benefits to having independent control of all six airplane states. This generally requires additional control surfaces and control manipulators, but once this capability is present there is a seemingly unlimited number of possible combinations of responses to a particular control input. At this point in the development of six-degree-of-freedom control capability, the response requirements for different tasks, limits on interference with other modes, etc. are not yet defined.

For unconventional and conventional modes the mechanical, electrical and hydraulic details of the closed-loop system between the pilot and the airplane response are totally irrelevant, at this stage we need only define the input-output relationship. Once the required aircraft responses are defined, the designer has the task of "inverting" these responses into the aerodynamic and system parameters that he can control. Note also that, until recently, a designer basically varied configuration parameters and the resulting stability and control derivatives. It was therefore logical to study the effects of these derivatives on flying qualities. With the current flight control capabilities to vary the dynamic responses in independent and unconventional ways, it is necessary to study the effects of the response variables directly and separately. This INVERSE PROBLEM requires the synergistic design of aerodynamic and equipment components to achieve adequate performance and responses which have been defined as desirable for each task.

In practice, this inverse problem is only partially addressed and the ground-based simulator is used to finalize the aircraft design. In order to satisfy that requirement, the simulation model has to be as representative of the aircraft system as possible. The current trend in design philosophy seems to be towards doing more and more functions with the flight control system (FCS). The F-16, for example, is a configuration in which the FCS has more authority than the pilot at certain flight conditions[3]. In general, modern aircraft have a proliferation of feedbacks, crossfeeds, feedforwards, filters, washouts, etc., resulting in very complex systems. A simulation model of such a system is, of necessity, also complex and its use for research is limited. On the other hand, we have seen that too much reliance on ground-based simulation for design use can lead to problems for the pilot in flight. We have a strong incentive, therefore, to develop general design criteria including simulation results validated by flight data. The following section details a simulation model which has been developed for that purpose.

II. Development of the Generic Model

Figure 1 is a very simplified layout of an aircraft simulation program. The heart of the program is the calculation of the linear and angular acceleration responses that result from the transient response and gust effects plus pilot inputs. In a conventional aircraft simulation these calculations occur in a block labeled "Equations of Motion", as indicated in Figure 2A. The form of the calculations may include table lookup, linear, nonlinear and cross-axis derivatives, etc. as indicated by the configuration data. For analysis purposes, it

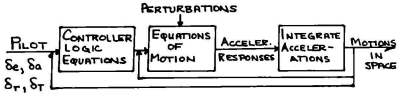


FIG. 1 SIMPLIFIED SIMULATION SCHEMATIC

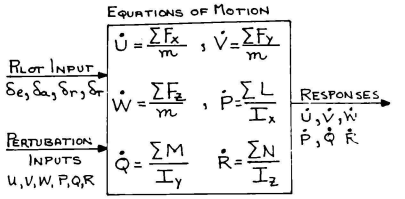


FIG. 2a CONVENTIONAL SIMULATION

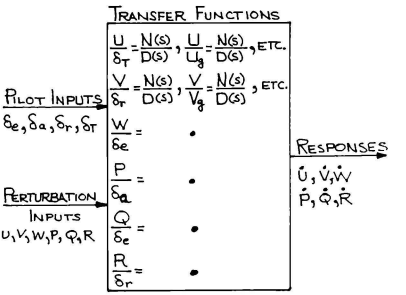


FIG. 2b GENERIC RESPONSE SIMULATION

is quite common to linearize the equations of motion about a point and to consider the input/output relationships as linear transfer functions. In principle then the simulation can be mechanized as in Figure 2B, provided we can reproduce the appropriate input/output relationships - that is the premise of the work to be described.

Another important factor is the concept of equivalent systems (References 6-8). By way of illustration, for a highly augmented aircraft, the pitch response to pilot stick force may be represented by a transfer function of high order, e.g.:

$$\frac{\dot{\theta}}{F_s} = K \frac{s^{33} + s^{32} + \ldots\ldots}{s^{34} + s^{33} + \ldots\ldots}$$

It is quite common to arrange for poles and zeros to cancel and there may also be terms which do not affect pilot control; even so the high-order dynamic system defies interpretation in most cases. We have found it convenient to approximate the frequency response of the high-order system with the response of a transfer function of classical, 'text-book form':

$$\frac{\dot{\theta}}{Fs} = \frac{K\,(S + 1/T_E)\; e^{AES}}{(s^2 + 2\zeta_E\,\omega_E\,S + \omega_E^2)}$$

Reference 1 requires the definition of such equivalent systems of classical form (with the addition of the equivalent time delay, A_E) for each response. The modal specification requirements are then applied to the equivalent system parameters. We have also seen that configuration with dynamics which cannot be matched by a classical equivalent system frequently have flying qualities problems. The apparent limitation of simple low-order transfer functions is not a problem for our purposes. The development of the model will start from these simple premise, add necessary nonlinear terms and remove the strict linear restrictions to achieve a flexible quasilinear model.

The Linear Model Derivation

In this paper, no attempt is made to relate specific controllers to aerodynamic surfaces or thrust devices. Instead the structure of the transfer functions is formed without concern for such physical details, we are only concerned with the overall response to pilot inputs. The "how" is relegated to the design process. It is convenient to think of the pilot/controller as a set of equations which feed into the generic model equations as illustrated in Figure 1.

The equations of motion for an aircraft, with symmetry about the x - z plane (Ixy = Iyz =0) are stated in terms of three linear accelerations:

$$m\begin{bmatrix}\dot{U}\\ \dot{V}\\ \dot{W}\end{bmatrix} = \begin{bmatrix}\Sigma X\\ \Sigma Y\\ \Sigma Z\end{bmatrix} + m\begin{bmatrix}0 & R & -Q\\ -R & 0 & P\\ Q & -P & 0\end{bmatrix}\begin{bmatrix}U\\ V\\ W\end{bmatrix} + mg\begin{bmatrix}-\sin\Theta\\ \cos\Theta\sin\Phi\\ \cos\Theta\cos\Phi\end{bmatrix} \quad (1)$$

plus three angular accelerations:

$$\begin{bmatrix}I_x & 0 & -I_{xz}\\ 0 & I_y & 0\\ -I_{xz} & 0 & I_z\end{bmatrix}\begin{bmatrix}\dot{P}\\ \dot{Q}\\ \dot{R}\end{bmatrix} = \begin{bmatrix}\Sigma L\\ \Sigma M\\ \Sigma N\end{bmatrix} + \begin{bmatrix}I_y - I_z & 0 & I_{xz}\\ 0 & I_z - I_x & 0\\ -I_{xz} & 0 & I_x - I_y\end{bmatrix}\begin{bmatrix}QR\\ RP\\ PQ\end{bmatrix} + \begin{bmatrix}0 & 0 & 0\\ 0 & I_{xz} & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}Q^2 - R^2\\ R^2 - P^2\\ P^2 - Q^2\end{bmatrix} \quad (2)$$

Implicit in equations (1) and (2) are the standard assumptions[4], such as 1) the airframe is a rigid body with no significant momentum, and 2) the earth is fixed and flat. For analysis purpose, equations (1) and (2) are linearized about a trim point and separated into longitudinal and lateral-directional equations again using standard assumptions[4] (e.g., perturbations are small from steady, level flight trim conditions). These linearized perturbed equations

are written as longitudinal equations:

$$\begin{bmatrix}\dot{u}\\ \dot{w}\\ \dot{q}\end{bmatrix}=\begin{bmatrix}dX/m\\ dZ/m\\ dM/I_y\end{bmatrix}+\begin{bmatrix}0 & -W_0 & 0\\ 0 & U_0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix}+g\theta\begin{bmatrix}1\\ 0\\ 0\end{bmatrix} \qquad (3)$$

and lateral-directional equations:

$$\begin{bmatrix}1 & 0 & 0\\ 0 & I_x & -I_{xz}\\ 0 & -I_{xz} & I_z\end{bmatrix}\begin{bmatrix}\dot{v}\\ \dot{p}\\ \dot{r}\end{bmatrix}=\begin{bmatrix}dY/m\\ dL\\ dN\end{bmatrix}+\begin{bmatrix}W_0 & 0 & -V_0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix}+g\varphi\begin{bmatrix}1\\ 0\\ 0\end{bmatrix} \qquad (4)$$

NOTE in these equations, the velocity couplings are removed. The inertia couplings and the gravity terms will be removed to form the response transfer functions and are re-introduced to form the complete equations of motion.

It is convenient at this point to consider the manipulation of aero forces and moments required to cast the response transfer functions in the desired format. The aerodynamic force and moment perturbations dx, dy, ---, dn are formed assuming; 1) zero variation of atmospheric properties for the small altitude perturbations and 2) first order aerodynamic derivatives, consistent with the small perturbation assumption. If λ represents a general perturbed state, the perturbations due to motion and atmoshperic disturbances are given by:

$$\lambda a = \lambda - \lambda g \qquad (5)$$

where λ is equal to the aircraft perturbation, a stands for "aerodynamic" (with respect to the instantaneous relative wind) and the subscript g identifies the perturbation due to a wind or gust component. Further, if δ represents a generalized control perturbation and T the generalized force or moment, the perturbation forces and moments are written as:

$$dT = \Sigma\left\{\frac{\partial T}{\partial \lambda}(\lambda - \lambda g) + \frac{\partial T}{\partial \dot{\lambda}}(\dot{\lambda} - \dot{\lambda} g)\right\} + \Sigma\frac{\partial T}{\partial \delta}\delta \qquad (6)$$

where:

$$\begin{aligned}\eta_k &= (X/m, Z/m, M/I_y)\\ \lambda_i &= (u, w, q)\\ \delta_i &= (\delta_u, \delta_w, \delta_q)\end{aligned} \qquad (7)$$

for the longitudinal equations, and:

$$\begin{aligned}\eta_k &= (Y/m, L, N)\\ \lambda_j &= (v, p, r)\\ \delta_j &= (\delta_v, \delta_p, \delta_r)\end{aligned} \qquad (8)$$

for the lateral-directional equations, with $\dot{\lambda}$ representing the time derivatives of the generalized states. The controllers are subscripted by the primary state each is intended to control and we have used the six controllers in anticipation of the final result.

Substitution of (6) into (3) and (4) results in the following force and moment perturbation expressions for the longitudinal and lateral-directional equations respectively:

$$[d\eta_k]=\left[\frac{\partial \eta_k}{\partial \lambda_i}\right][\lambda_i-\lambda_{g_i}]+\left[\frac{\partial \eta_k}{\partial \dot{\lambda}_i}\right][\dot{\lambda}_i-\dot{\lambda}_{g_i}]+\left[\frac{\partial \eta_k}{\partial \delta_i}\right][\delta_i] \qquad (9)$$

$$[d\eta_k]\left[\frac{\partial \eta_k}{\partial \lambda_j}\right][\lambda_j-\lambda_{g_j}]+\left[\frac{\partial \eta_k}{\partial \dot{\lambda}_j}\right][\dot{\lambda}_j-\dot{\lambda}_{g_j}]+\left[\frac{\partial \eta_k}{\partial \delta_j}\right][\delta_j] \qquad (10)$$

where the $\partial(\ ,\ ,\ )/\partial(\ ,\ ,\ )$ are the Jacobian matricies of the force/moment with respect to the states or controllers. Equation (4) is seen to contain a set of equations that are coupled in $\dot{p}$ and $\dot{r}$. These two equations may be solved simultaneously as follows to form $\dot{p}$ and $\dot{r}$ with inertial coupling terms revoved:

$$\begin{bmatrix}\dot{p}\\ \dot{r}\end{bmatrix}=[MOI]^{-1}\left\{\left[\frac{\partial \eta_k^*}{\partial \lambda_j}\right][\lambda_j-\lambda_{g_j}]+\left[\frac{\partial \eta_k^*}{\partial \dot{\lambda}_j}\right][\dot{\lambda}_j-\dot{\lambda}_{g_j}]+\left[\frac{\partial \eta_k^*}{\partial \delta_j}\right][\delta_j]\right\} \qquad (11)$$

where for this set of equations, the partial derivatives of T^* are:

$$\frac{\partial \eta_k^*}{\partial \lambda_j}=\frac{\partial(L,N)}{\partial(v,p,r)} := L_{\lambda_j} \text{ OR } N_{\lambda_j} \qquad (12)$$

$$\frac{\partial \eta_k^*}{\partial \delta_j}=\frac{\partial(L,N)}{\partial(\delta_v,\delta_p,\delta_r)} := L_{\delta_j} \text{ OR } N_{\delta_j} \qquad (13)$$

$$[MOI]^{-1}=\begin{bmatrix}I_x & -I_{xz}\\ -I_{xz} & I_z\end{bmatrix}^{-1}=\frac{1}{det}\begin{bmatrix}I_z & I_{xz}\\ I_{xz} & I_x\end{bmatrix} \qquad (14)$$

$$det = I_x I_z - I_{xz}^2 \qquad (15)$$

Substituting (12) through (15) into (11) and carrying out the required algebra yields:

$$\begin{bmatrix}\dot{p}\\ \dot{r}\end{bmatrix}=\frac{1}{det}\begin{bmatrix}\Sigma\{I_z L_{\lambda_j}+I_{xz}N_{\lambda_j}\}(\lambda_j-\lambda_{g_j})\\ \Sigma\{I_{xz}L_{\lambda_j}+I_x N_{\lambda_j}\}(\lambda_j-\lambda_{g_j})\end{bmatrix} \qquad (16)$$

$$+\frac{1}{d\omega t}\begin{bmatrix}\Sigma\{I_Z L_{\lambda_j} + I_{XZ} N_{\lambda_j}\}(\lambda_j - \dot{\lambda}_{\dot{\delta}_j}) \\ \Sigma\{I_{XZ} L_{\lambda_j} + I_X N_{\lambda_j}\}(\lambda_j - \dot{\lambda}_{\dot{\delta}_j})\end{bmatrix}$$

$$+\frac{1}{d\omega t}\begin{bmatrix}\Sigma\{I_Z L_{\delta_j} + I_{XZ} N_{\delta_j}\}\delta_j \\ \Sigma\{I_{XZ} L_{\delta_j} + I_X N_{\delta_j}\}\delta_j\end{bmatrix} \qquad (16)$$

Equation (16) is evaluated at the operating points in steady flight in a homogenous atmosphere as we have assumed, and produces a set of constants. If we further define for the j th state or controller:

$$\frac{\partial L'}{\partial \lambda_j} = \frac{1}{d\omega t}\left[I_Z \partial L/\partial \lambda_j + I_{XZ}\partial N/\partial \lambda_j\right] \qquad (17)$$

and,

$$\frac{\partial N'}{\partial \lambda_j} = \frac{1}{d\omega t}\left[I_{XZ}\partial L/\partial \lambda_j + I_X \partial N/\partial \lambda_j\right] \qquad (18)$$

with analogous definitions for $\partial L'/\partial \dot{\lambda}j$, $\partial N'/\partial \dot{\lambda}j$, $\partial L'/\partial \dot{\delta}j$ and $\partial N'/\partial \dot{\delta}j$, equation (11) may be rewritten as,

$$\begin{bmatrix}\dot{p}\\ \dot{r}\end{bmatrix} = \left[\frac{\partial(L',N')}{\partial \lambda_j}\right][\lambda_j - \lambda_{\dot{\delta}_j}] + \left[\frac{\partial(L',N')}{\partial \dot{\lambda}_j}\right][\dot{\lambda}_j - \dot{\lambda}_{\ddot{\delta}_j}] + \left[\frac{\partial(L',N')}{\partial \dot{\delta}_j}\right][\dot{\delta}_j] \qquad (19)$$

where $\partial(L', N')/\partial\lambda j = \partial(L', N')/\partial(v, p, r)$, etc..

The lateral-directional equations given by (4) are rewritten as:

$$\begin{bmatrix}\dot{v}\\ \dot{p}\\ \dot{r}\end{bmatrix} = \left[\frac{\partial T_k''}{\partial \lambda_j}\right][\lambda_j - \lambda_{\dot{\delta}_j}] + \left[\frac{\partial T_k''}{\partial \dot{\lambda}_j}\right][\dot{\lambda}_j - \dot{\lambda}_{\ddot{\delta}_j}] + \left[\frac{\partial T_k''}{\partial \dot{\delta}_j}\right][\dot{\delta}_j]$$

$$+\begin{bmatrix}W_0 & 0 & -V_0\\ 0 & 0 & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} - g\varphi\begin{bmatrix}1\\ 0\\ 0\end{bmatrix} \qquad (20)$$

with the derivative of T^* defined as follows:

$$\frac{\partial T_k''}{\partial \lambda_j} = \frac{\partial(Y/m, L', N')}{\partial(v, p, r)} \qquad (21)$$

$$\frac{\partial T_k''}{\partial \dot{\lambda}_j} = \frac{\partial(Y/m, L', N')}{\partial(\dot{v}, \dot{p}, \dot{r})} \qquad (22)$$

$$\frac{\partial T''}{\partial \delta_j} = \frac{\partial(Y/m, L', N')}{\partial(\delta v, \delta p, \delta r)} \qquad (23)$$

Equations (3) and (20) are the perturbed longitudinal and lateral-directional equations used to form the generic transfer functions. Taking the Laplace transform of (3) and (20) and rearranging results in the following sets:

Longitudinal,

$$\begin{bmatrix}s(1-X_{\dot{u}})-X_u & -(sX_{\dot{w}}+X_w) & -(sX_{\dot{q}}+X_q)+W_0+\frac{g}{s}\\ -(sZ_{\dot{u}}+Z_u) & s(1-Z_{\dot{w}})-Z_w & -(sZ_{\dot{q}}+Z_q)-U_0\\ -(sM_{\dot{u}}+M_u) & -(sM_{\dot{w}}+M_w) & s(1-M_{\dot{q}})-M_q\end{bmatrix}\begin{bmatrix}u\\ w\\ q\end{bmatrix} =$$

$$-\begin{bmatrix}sX_{\dot{u}}+X_u & sX_{\dot{w}}+X_w & sX_{\dot{q}}+X_q\\ sZ_{\dot{u}}+Z_u & sZ_{\dot{w}}+Z_w & sZ_{\dot{q}}+Z_q\\ sM_{\dot{u}}+M_u & sM_{\dot{w}}+M_w & sM_{\dot{q}}+M_q\end{bmatrix}\begin{bmatrix}u_g\\ w_g\\ q_g\end{bmatrix} \qquad (24)$$

$$+\begin{bmatrix}X_{\delta u} & X_{\delta w} & X_{\delta_e}\\ Z_{\delta u} & Z_{\delta w} & Z_{\delta_e}\\ M_{\delta u} & M_{\delta w} & M_{\delta_e}\end{bmatrix}\begin{bmatrix}\delta u\\ \delta w\\ \delta_e\end{bmatrix}$$

Lateral-Directional,

$$\begin{bmatrix}s(1-Y_{\dot{v}})-Y_v & -(sY_{\dot{p}}+Y_p)-W_0+\frac{g}{s} & -(sY_{\dot{r}}+Y_r)+V_0\\ -(sL'_{\dot{v}}+L'_v) & s(1-L'_{\dot{p}})-L'_p & -(sL'_{\dot{r}}+L'_r)\\ -(sN'_{\dot{v}}+N'_v) & -(sN'_{\dot{p}}+N_p) & s(-N'_{\dot{r}})-N'_r\end{bmatrix}\begin{bmatrix}v\\ p\\ r\end{bmatrix} =$$

$$-\begin{bmatrix}sY_{\dot{v}}+Y_v & sY_{\dot{p}}+Y_p & sY_{\dot{r}}+Y_r\\ sL'_{\dot{v}}+L'_v & sL'_{\dot{p}}+L'_p & sL'_{\dot{r}}+L'_r\\ sN'_{\dot{v}}+N'_v & sN'_{\dot{p}}+N'_p & sN'_{\dot{r}}+N'_r\end{bmatrix}\begin{bmatrix}v_g\\ p_g\\ r_g\end{bmatrix} \qquad (25)$$

$$+\begin{bmatrix}Y_{\delta v} & Y_{\delta_a} & Y_{\delta_r}\\ L'_{\delta v} & L'_{\delta_a} & L'_{\delta_r}\\ N'_{\delta v} & N'_{\delta_a} & N'_{\delta_r}\end{bmatrix}\begin{bmatrix}\delta v\\ \delta_a\\ \delta r\end{bmatrix}$$

The aircraft longitudinal and lateral-directional transfer functions, $u/\delta w$, $u/\delta q$, ..., and $p/\delta p$, $p/\delta r$, ..., are obtained from equations (6) and (25). The velocity coupling terms have already been removed by linearization of the equations of motion. Since gravity occurs in the u-equation only, the conventional short-period assumption removes it from the linear representation. We recognize that it needs to be included in the final overall formulation. Assuming $U_o \approx 0$ the longitudinal motion occurs in w and q and equation (24) reduces to:

$$\begin{bmatrix} s(1-Z_{\dot w})-Z_w & -(sZ_{\dot q}+Z_q+U_o) \\ -(sM_{\dot w}+M_w) & s(1-M_{\dot q})-M_q \end{bmatrix} \begin{bmatrix} w \\ q \end{bmatrix} =$$

$$- \begin{bmatrix} sZ_{\dot w}+Z_w & sZ_{\dot q}+Z_q \\ sM_{\dot w}+M_w & sM_{\dot q}+M_q \end{bmatrix} \begin{bmatrix} w_g \\ q_g \end{bmatrix} \qquad (26)$$

$$+ \begin{bmatrix} Z_{\delta w} & Z_{\delta q} \\ M_{\delta w} & M_{\delta q} \end{bmatrix} \begin{bmatrix} \delta_w \\ \delta_q \end{bmatrix}$$

which produces the following transfer functions using Cramers Rule:

$$\frac{w}{\delta_w} = \frac{|N^w_{\delta w}|}{|D_L|} \qquad \frac{w}{\delta_q} = \frac{|N^w_{\delta q}|}{|D_L|} \qquad (27)$$

$$\frac{q}{\delta_w} = \frac{|N^q_{\delta w}|}{|D_L|} \qquad \frac{q}{\delta_q} = \frac{|N^q_{\delta q}|}{|D_L|} \qquad (28)$$

$$\frac{w}{w_g} = \frac{|N^w_{w_g}|}{|D_L|} \qquad \frac{w}{q_g} = \frac{|N^w_{q_g}|}{|D_L|} \qquad (29)$$

$$\frac{q}{w_g} = \frac{|N^q_{w_g}|}{|D_L|} \qquad \frac{q}{q_g} = \frac{|N^q_{q_g}|}{|D_L|} \qquad (30)$$

where,

$$|D_L| = \begin{vmatrix} s(1-Z_{\dot w})-Z_w & -(sZ_{\dot q}+Z_q+U_o) \\ -(sM_{\dot w}+M_w) & s(1-M_{\dot q})-M_q \end{vmatrix}$$

$$= s^2\left\{(1-Z_{\dot w})(1-M_{\dot q})-M_{\dot w}Z_{\dot q}\right\} \qquad (31)$$

$$+s\left\{Z_w(1-M_{\dot q})+M_q(1-Z_{\dot w})+M_w Z_{\dot q}\right.$$

$$\left.+M_{\dot w}(Z_q+U_o)\right\}+M_q Z_w - M_w(Z_q+U_o)$$

and the numerators, $N^{\lambda}_{\delta j}$ or λ_j, are obtained by replacing the column of λ coefficients in D_L by the column of δ_j or λ_j coefficients from the right hand side of Eq. (26); e.g.,

$$|N^w_{\delta w}| = \begin{vmatrix} Z_{\delta w} & -(sZ_{\dot q}+Z_q+U_o) \\ M_{\delta w} & s(1-M_{\dot q})-M_q \end{vmatrix} \qquad (32)$$

$$= s\left[Z_{\delta w}(1-M_{\dot q})+M_{\delta w}Z_{\dot q}\right]$$

$$+M_{\delta w}(Z_q+U_o)-Z_{\delta w}M_q$$

Equations (27) to (30) provide 8 of 18 transfer functions possible form the conventional aircraft longitudinal equations. Six of the eighteen transfer functions relate speed changes to controller or "gusts".

In many flying qualities applications we are considering short term pilot control, which is consistent with the short-period assumptions. For correctness, the nonlinear gravity term is added back into the final formulation, restoring a phugoid (see Ref. 9). This is then consistent with linear velocity effects modeled by a first order transfer function, e.g.,

$$\frac{u}{\delta_u} = \frac{K^u_{\delta u}}{A^u_{\delta u}s + B^u_{\delta u}} \qquad (33)$$

where K, A, B are constants with the super and subscripts denoting, respectively, the state and controller to which the constants are associated. Similar expressions can be written for $u/\delta w$, $u/\delta q$, u/w_g and u/q_g. Discussion of the four transfer functions $w/\delta u$, w/u_g, $q/\delta u$ and q/u_g is defered until after the lateral-directional response equations are obtained.

For the lateral-directional transfer functions, the roll mode is often assumed to be a one degree of freedom motion with v and r suppressed. This gives;

$$\left[s(1-L'_{\dot p})-L'_p\right]p = -(sL'_{\dot p}+L'_p)p_g + L'_{\delta_p}\delta_p \qquad (34)$$

$$\frac{p}{\delta_p} = \frac{L'_{\delta_p}}{s(1-L'_{\dot p})-L'_p} \qquad (35)$$

$$\frac{p}{p_g} = \frac{-(sL'_{\dot p}+L'_p)}{s(1-L'_{\dot p})-L'_p} \qquad (36)$$

This implies that the influence of the Dutch roll mode has been cancelled by a similar term in the numerator; however, the more general form is used in the next section.

The Dutch roll mode is a two-degree of freedom

motion in either v and r with $p \approx o$, or r and p with $v \approx o$, depending on the configuration. The generalized form can account for either possibility; as an example the first case yields:

$$\begin{bmatrix} s(1-Y_{\dot{v}})-Y_v & -(sY_{\dot{r}}+Y_r) \\ -(sN'_{\dot{v}}+N'_v) & s(1-N'_{\dot{r}})-N'_r \end{bmatrix} \begin{bmatrix} v \\ r \end{bmatrix} = -\begin{bmatrix} sY_{\dot{v}}+Y_v & sY_{\dot{r}}+Y_r \\ sN'_{\dot{v}}+N'_v & sN'_{\dot{r}}+N'_r \end{bmatrix} \begin{bmatrix} v_g \\ r_g \end{bmatrix}$$

$$+\begin{bmatrix} Y_{\delta v} & Y_{\delta r} \\ N'_{\delta v} & N'_{\delta r} \end{bmatrix} \begin{bmatrix} \delta v \\ \delta r \end{bmatrix} \quad (37)$$

With the transfer functions defined as;

$$\frac{v}{\delta v} = \frac{|N^v_{\delta v}|}{|D_D|} \qquad \frac{v}{\delta r} = \frac{|N^v_{\delta r}|}{|D_D|} \quad (38)$$

$$\frac{r}{\delta v} = \frac{|N^r_{\delta v}|}{|D_D|} \qquad \frac{r}{\delta r} = \frac{|N^r_{\delta r}|}{|D_D|} \quad (39)$$

$$\frac{v}{v_g} = \frac{|N^v_{v_g}|}{|D_D|} \qquad \frac{v}{r_g} = \frac{|N^v_{r_g}|}{|D_D|} \quad (40)$$

$$\frac{r}{v_g} = \frac{|N^r_{v_g}|}{|D_D|} \qquad \frac{r}{r_g} = \frac{|N^r_{r_g}|}{|D_D|} \quad (41)$$

where,

$$|D_D| = \begin{vmatrix} s(1-Y_{\dot{v}})-Y_v & -(sY_{\dot{r}}+Y_r) \\ -(sN'_{\dot{v}}+N'_v) & s(1-N'_{\dot{r}})-N'_r \end{vmatrix} \quad (42)$$

the remaining undefined 4 longitudinal and 8 lateral-directional transfer functions are obtained using the following observation. If equations (24) and (25) are examined, it becomes obvious that transfer functions may be formed using the full 3 X 3 determinants of the matrices. The controller and "gust" numerators are all 2nd and 3rd order respectively with 3rd order denominators. What is important is that the maximum order of the controller numerators are the same as are the gust numerators and both sets of denominators.

Using this observation, the remaining transfer functions have the following size;

longitudinal mode,

w/δu and q/δu are 1st/2nd order,
w/ug and q/ug are 2nd/2nd order,

lateral-directional mode,

v/δv and r/δv are 1st/2nd order,
v/vg and r/vg are 2nd/2nd order,
p/δv and p/δr are oth/1st order,
p/vg and p/rg are 1st/1st order,

The explicit symbolic statement of these functions are given in the next section.

Transfer Function Generalization

In the past, aircraft responses could be adequately described by low-order mathematical models. These models included, for example, the longitudinal short-period approximation and the first-order roll rate response approximation. They were not true simulations of the dynamics, which were higher order, but the region over which the approximations were valid was understood and the simple models were useful in design and analysis. For this reason, their parameter values are also used for formal specification of flying qualities in MIL-F-8785C (Ref 1). Their validity is so well understood that there is barely any reference in the specification to the linearization and decoupling needed to establish the longitudinal short-period approximation. Thus, the process of order reduction is well known to flying qualities analysis. Together with the equivalent system technique, it is therefore a logical starting point for generic simulation analysis of high-order effects introduced by feedback control systems, augmentation, etc.

References 6 through 8 provide three examples of work exploring the use of equivalent systems and what effects various terms have on pilot opinion. These and other studies provide strong evidence that equivalent systems are an effective and valid approach to modeling high-order aircraft dynamics. Conculsions that have been drawn from these studies are:

> The response of many high-order systems can be matched closely for conventional and augmented/control configured aircraft. Time delays provide a good representation of high-frequency phase lag and digital sampling computation delays. Both longitudinal and lateral-directional response time delays are necessary to account for lag introduced by high-frequency components.

In order to allow sufficient flexibility,

however, we consider additional terms in both numerator and denominator to give a general form:

$$\frac{\lambda}{\delta} = \frac{Ke^{-As}\ (T_E s + 1)}{(s^2 + 2\xi\omega s + \omega^2)} \cdot \frac{(t_1 s + 1)}{(t_2 s + 1)} \qquad (43)$$

The advantage of equation (43) is the equation parameters can be related to familiar features of the aircraft and flight control system. For example, the relation of frequency, ω, damping ratio, ξ , and numerator time constant, T_E, to vehicle shape, loading, flight conditions and feedback control laws is documented in Reference 5. The lead and lag contributions of flight control system components such as actuators, sensors and filters are similarly known. Digital control systems are historically less familiar but their contributions to lag are now being realized.

For example the equation for u/δu becomes;

$$\frac{u}{\delta_u} = \frac{K^u_{\delta u} e^{-C^u_{\delta u} s}}{A^u_{\delta u} s + B^u_{\delta u}} \cdot \frac{D^u_{\delta u} s + E^u_{\delta u}}{F^u_{\delta u} s + G^u_{\delta u}} \qquad (44)$$

where $A^u_{\delta u}$, ..., $G^u_{\delta u}$ are constants associated with u/δu. The exponential term in Eq (44) is simplified using the first order Pade' approximation and is rewritten as;

(45)

$$\frac{u}{\delta_u} = \frac{K^u_{\delta u}}{A^u_{\delta u} s + B^u_{\delta u}} \cdot \frac{1 - C^u_{\delta u} s}{1 + C^u_{\delta u} s} \cdot \frac{D^u_{\delta u} s + E^u_{\delta u}}{F^u_{\delta u} s + G^u_{\delta u}}$$

In a similar manner the remaining transfer functions for controllers and "gust" can be generalized.

The most general representation of aircraft responses to controllers and gusts can be written as one matrix equation as follows;

$$\begin{bmatrix} u \\ v \\ \omega \\ \phi \\ q \\ r \end{bmatrix} = \begin{bmatrix} u/\delta_u & u/\delta_v & u/\delta_\omega & u/\delta_\phi & u/\delta_q & u/\delta_r \\ v/\delta_u & v/\delta_v & v/\delta_\omega & v/\delta_\phi & v/\delta_q & v/\delta_r \\ \omega/\delta_u & \omega/\delta_v & \omega/\delta_\omega & \omega/\delta_\phi & \omega/\delta_q & \omega/\delta_r \\ \phi/\delta_u & \phi/\delta_v & \phi/\delta_\omega & \phi/\delta_\phi & \phi/\delta_q & \phi/\delta_r \\ q/\delta_u & q/\delta_v & q/\delta_\omega & q/\delta_\phi & q/\delta_q & q/\delta_r \\ r/\delta_u & r/\delta_v & r/\delta_\omega & r/\delta_\phi & r/\delta_q & r/\delta_r \end{bmatrix} \begin{bmatrix} \delta_u \\ \delta_v \\ \delta_\omega \\ \delta_\phi \\ \delta_q \\ \delta_r \end{bmatrix} \qquad (46)$$

$$+ \begin{bmatrix} u/u_g & u/v_g & u/\omega_g & u/\phi_g & u/q_g & u/r_g \\ v/u_g & v/v_g & v/\omega_g & v/\phi_g & v/q_g & v/r_g \\ \omega/u_g & \omega/v_g & \omega/\omega_g & \omega/\phi_g & \omega/q_g & \omega/r_g \\ \phi/u_g & \phi/v_g & \phi/\omega_g & \phi/\phi_g & \phi/q_g & \phi/r_g \\ q/u_g & q/v_g & q/\omega_g & q/\phi_g & q/q_g & q/r_g \\ r/u_g & r/v_g & r/\omega_g & r/\phi_g & r/q_g & r/r_g \end{bmatrix} \begin{bmatrix} u_g \\ v_g \\ \omega_g \\ \phi_g \\ q_g \\ r_g \end{bmatrix}$$

In general, many of these terms will be zero for conventional or unconventional aircraft. In addition, a given simulation experiment would probably include many zero terms to bound the scope of the task.

Equation (46) can also be written as:

$$\lambda = A\delta + G\lambda g \qquad (47)$$

with,

$$\lambda = \begin{bmatrix} u \\ v \\ \omega \\ \phi \\ q \\ r \end{bmatrix} \qquad \delta = \begin{bmatrix} \delta_u \\ \delta_v \\ \delta_\omega \\ \delta_\phi \\ \delta_q \\ \delta_r \end{bmatrix} \qquad (48)$$

and,

$$A = \left[a_{ij}\right] \qquad (49)$$

$$G = \left[g_{ij}\right] \qquad (50)$$

The individual transfer functions contained in the controller and gust matrices, A and G, can be conveniently written as follows:

$$a_{ij} = M_{ij} / d_{ij} \qquad (51)$$

$$d_{ij} = (\varepsilon_{ij} s^2 + \xi_{ij} s + \eta_{ij})(\lambda_{ij} s + \mu_{ij})(f_{ij} s + \sigma_{ij}) \qquad (52)$$

$$M_{ij} = K_{ij}(A_{ij} s + B_{ij})(C_{ij} s + D_{ij})(E_{ij} s + F_{ij})$$

and,

$$g_{ij} = N_{ij} / d_{ij} \qquad (54)$$

$$N_{ij} = k_{ij}(\alpha_{ij} s + \beta_{ij})(\delta'_{ij} s + \theta_{ij})(\gamma'_{ij} s + \phi_{ij}) \qquad (55)$$

For the a_{11} element, we have from eqs (51) to (55), the following correspondence to $u/\delta u$:

$$a_{11} = u/\delta u$$
$$= \frac{K_{11}(C_{11}s + D_{11})(E_{11}s + F_{11})}{(\xi_{11}s + \eta_{11})(\lambda_{11}s + \mu_{11})(\beta_{11}s + \sigma_{11})}$$

with,

$$K_{11} = K^u_{\delta u}\ ,\ A_{11} = 0\ ,\ B_{11} = 1\ ,\ C_{11} = -C^u_{\delta u}\ ,\ D_{11} = 1\ ,$$
$$E_{11} = D^u_{\delta u}\ ,\ F_{11} = E^u_{\delta u}\ ,\ \beta_{11} = 0\ ,\ \xi_{11} = A^u_{\delta u}\ ,\ \eta_{11} = B^u_{\delta u}\ ,$$
$$\lambda_{11} = C^u_{\delta u}\ ,\ \mu_{11} = 1\ ,\ \rho_{11} = F^u_{\delta u}\ ,\ \sigma_{11} = G^u_{\delta u}$$

when $u/\delta u$ is given by eq (44). Similar relations can be defined for each element of $A=[a_{ij}]$ and $G=[g_{ij}]$; e.g., a set of relations can be defined using equations (27) through (30) and (51) to (55). Equations (51) and (54) can also be defined directly in terms of lumped parameters, such as natural frequency and damping, time delays and lead-lags which directly affect the aircraft dynamics and flying qualities. Putting these equations on a simulator, with the input being the relevant response parameters, provides a dynamic model useful to the flying qualities researcher. One limitation implied by the use of transfer functions to represent responses to controller inputs is that the coefficients appearing in equations (51) and (54) are constant. This assumption appeared when the initial conditions were set equal to zero (trim points). This assumption was useful during the subsequent analysis; however, its impact can be reduced by scheduling the coefficients as functions of velocity or other parameters as appropriate for the particular task at hand. To see how this can be done, we will examine the coefficients in the quadratic denominator of the longitudinal set of transfer functions, equations (27) through (30). This can be expressed as:

$$D_L = s^2 + 2\omega_{sp}\,\xi_{sp}\,s + \omega^2_{sp} \qquad (56)$$

The two coefficients ω_{sp} and ξ_{sp} are defined in classical short-period form as functions of aerodynamic variables[4] :

$$W_{sp} = (M_q Z_w - V_a M_w)^{1/2} \qquad (57)$$
$$\xi_{sp} = -(V_a M_w + Z_w + M_q)/2W_{sp} \qquad (58)$$

where Va ~ velocity relative to air mass,

$$M_q \propto C_1 V_a$$
$$Z_w \propto C_2 V_a$$
$$M_w \propto C_3 V_a$$
$$M_{\dot{w}} \propto C_4 V_a \qquad (59)$$
$$M_\delta \propto C_5 V_a$$

and C_1, C_2, ..., C_5 are constants for this discussion, and are associated with the four aerodynamic parameters listed above.

Substituting definitions given by (59) into eqs (57) and (58) we get;

$$\omega_{sp} = (C_1 V_a C_2 V_a - V_a C_3 V_a)^{1/2} \qquad (60)$$
$$\propto V_a \qquad (61)$$

and,

$$\xi_{sp} = (V_a C_4 + C_2 V_a + C_1 V_a)/2V_a \qquad (62)$$

i.e. invariant with airspeed

The "constants" in equation (56) may be scheduled with short-period frequency proportional to airspeed and short-period damping ratio constant to represent the effects of airspeed variations about the trim point.

A similar analysis can be performed on all the parameters appearing in the numerators and denominators of equations (51) and (54). An advantage of using the transfer function model is the parameter scheduling can be defined by continuous or piece-wise continuous polynomial functions instead of tabular data and table look up algorithms usually required for aerodynamic/mass/engine data. Also, in principle we can schedule the response characteristics with any variable such as angle of attack, Mach number, etc.

Although the preceding discussion has been concerned with conventional aircraft, we have included six controls in the equations. In its most general form, equation(46) can apply to six-degree-of-freedom control configurations including interference of each control in other axes. We also have the option of redefining the state variables to be controlled, for example flight path angle instead of angle of attack. In this way we can study any dynamic response which is amenable to linear analysis, plus many in a quasinonlinear fashion.

III. Complete Acceleration Model Equations

With the aircraft response to controller and gust response equations defined, the complete acceleration equations may be written. This is done by adding to the linear response equations, gravity and velocity coupling effects. Depending on the particular experiment to be done, some of the nonlinear terms may be insignificant. If this situation occurs, we

want the ability to zero out those terms through the simulation input. Consequently, each nonlinear term is premultiplied by a constant that may be set to a zero or non-zero value. In this manner any of the nonlinear terms may be used in the simulation model. The acceleration equations for translation and rotational motion for a conventional or "six degree of freedom" aircraft are written as follows:

$$\begin{bmatrix}\dot{U}\\ \dot{V}\\ \dot{W}\end{bmatrix} = [a_{ij}]\begin{bmatrix}\delta_u\\ \delta_v\\ \delta_w\\ \delta_P\\ \delta_Q\\ \delta_R\end{bmatrix} + [g_{ij}]\begin{bmatrix}U_g\\ V_g\\ W_g\\ P_g\\ Q_g\\ R_g\end{bmatrix} + \begin{bmatrix}0 & e_{12}R & e_{13}Q\\ e_{21}R & 0 & e_{23}P\\ -e_{31}Q & -e_{32}P & 0\end{bmatrix}\begin{bmatrix}U\\ V\\ W\end{bmatrix} \tag{63}$$

$$+ g\begin{bmatrix}-f_{11}\sin\Theta\\ f_{21}\cos\Theta\sin\Phi\\ f_{31}\cos\Theta\cos\Phi\end{bmatrix}$$

for i = 1, 2, 3, and j = 1, ..., 6.

$$\begin{bmatrix}I_x & 0 & -I_{xz}\\ 0 & I_y & 0\\ -I_{xz} & 0 & I_z\end{bmatrix}\begin{bmatrix}\dot{P}\\ \dot{Q}\\ \dot{R}\end{bmatrix} = [a_{ij}]\begin{bmatrix}\delta_u\\ \delta_v\\ \delta_w\\ \delta_P\\ \delta_Q\\ \delta_R\end{bmatrix} + [g_{ij}]\begin{bmatrix}U_g\\ V_g\\ W_g\\ P_g\\ Q_g\\ R_g\end{bmatrix} \tag{64}$$

$$+ \begin{bmatrix}I_y-I_z & 0 & I_{xz}\\ 0 & I_z & I_x\\ -I_{xz} & 0 & I_x-I_y\end{bmatrix}\begin{bmatrix}QR\\ RP\\ PQ\end{bmatrix} + \begin{bmatrix}0 & 0 & 0\\ 0 & I_{xz} & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}Q^2-R^2\\ R^2-P^2\\ P^2-Q^2\end{bmatrix}$$

for i = 4, 5, 6, and j = 1, ..., 6. The matrices A_{ij} and C_{ij} are given in sections II or III. It is noted that equations (63) and (64) are written in mixed notion (time and frequency domain) for convenience.

Not all the assumptions made in section II and III are in the final generic simulation acceleration equations. They are not separated into longitudinal and lateral-directional sets, the transfer function coefficients may be scheduled, and inertial coupling and gravity have been re-introduced. Furthermore, by appropriate selection of the transfer function parameters, equations (63) and (64) may be used for conventional and non-convention (i.e. six-degree-of-freedom) aircraft simulations. With transfer function terms representing interference of primary control coupling with secondary control effects, these equations are suited for initial 6 DOF flying qualities experiments.

IV. Proposed Simulation Validation

In order for the generic simulation to have any credibility, we must verify that: 1) to the pilot it seems to "fly like an airplane" and 2) it gives the correct trends in the effects of response variables. The first part will be satisfied by careful check of the mechanization of linear and nonlinear terms. The second part will be accomplished by comparison with flying qualities results from recent in-flight simulation programs (Ref 10 through 13). These programs were conducted on variable stability airplanes, the NT-33 and the C-131 Total In-Flight Simulator (TIFS). The results are in the form of pilot ratings[14] and pilot comments over ranges of aircraft response characteristics for different piloting tasks.

Reference 10 is the 'Neal-Smith report', well-known to flying qualities specialists. It contains results for up-and-away flying of simulated configurations with a wide variety of longitudinal short-period characteristics. Various lead and lag terms were also added to the basic short-period modal characteristics to represent the effects of control system dynamics. The short-period frequencies and damping ratios are representative of fighter configurations and have a range of Level 1/Level 2 flying qualities as defined by MIL-F-8785C. The evaluation maneuvers were designed to be representative of air-to-air combat and air-to-ground weapon delivery, including both visual and instrument flight. Reference 11 contains the results of a similar experiment conducted for the landing approach task, including actual touchdowns on the runway in the evaluation configurations. Ranges of values of roll mode and lead/lag filter time constants, and Dutch roll damping ratios were also evaluated in the NT-33 for various tasks (Ref 12).

Reference 10 - 12 therefore contain sufficient data to validate all axes of the generic model for a wide range of characteristics appropriate to fighter configurations. The conditions of the different in-flight experiments will be replicated as closely and as reasonably possible on the Flight Dynamics Laboratory ground-based simulation facility. Appropriate tasks will be 'flown' by pilots and the ratings and comments will be compared with those obtained in flight. Exact correspondence is not to be expected, since there are inconsistencies and scatter in the NT-33 results. There are also the obvious differences in cues, cockpit, etc. between the ground-based and in-flight experiments. We do expect, however, to verify the generic simulation concept and the data trends, or analyze and explain any consistent differences between the data sets.

Note that the generic formulation being used included the classical modal representations, plus first-order lead and lag terms. NT-33 configurations having this form will be repeated in this form and also with the lead/lag effects included in a equivalent transfer function of classical form. For further discussion of the equivalent system approach see References 6 and 7.

If the equivalent system approach to flying qualities is valid, then the different formulations should appear identical to the pilot in terms of task performance and pilot ratings. This expected result will also validate the concept of using the generic simulation formulation as an equivalent system to represent complex, high-order aircraft configurations.

The preceding results will lead to the next step in validating the simulation for large aircraft using the results of a recent in-flight program on TIFS (Ref 13). In this experiment, the effects of both longitudinal and lateral-configuration and augmentation parameters were assessed for the landing approach task. In general, replication of a range of conditions from the four in-flight simulations cited, and correlation of the results, is expected to provide a complete validation of the generic simulation for conventional aircraft plus the experience necessary to proceed with meaningful flying qualities research.

A recent contracted effort (Ref 15), to develop criteria for the response of direct force control modes included some in-flight simulation of the use of direct side-force control. These results, plus results from ground-based simulations conducted in the Flight Dynamics Laboratory, will be used to validate the use of the generic simulation for unconventional flight control modes.

The main incentive for formulating the generic simulation model was the difficulty of obtaining the required parameter variations in a typical system simulation. The validation experiments listed in the preceding section will also represent the first research experiment in expanding the data bases from the in-flight simulations and explaining any anomalies between the two sets of results. We may encounter differences, possibly as functions of airplane class, piloting task, simulator motion cues, etc., which must be analysed and explained before the simulation model is considered valid and useful for research in other areas.

V. Uses of the Generic Model

Once the model has been validated, we then have a 'shopping list' of topics that will be addressed:

1) requirements for independent control of six degrees of freedom, including evaluation of controllers,

2) requirements for STOL/low speed control, especially for fighter configurations,

3) application of equivalent system requirements to unconventional configurations,

4) sidestick controller requirements for transport configurations

5) Atmospheric disturbance effects - verification and development of the MIL-F-8785C model,

6) study of interactions between aircraft response dynamics and display dynamics.

Mini-experiments on these and other topics will be conducted, with the generic model defined as consistent with the different objectives. Each of the terms in Equation (63) and (64)will be tailored according to the experiment.

The generic model also has potential applications in areas not directly associated with manned simulations. Specifically, we see the model used in analytical studies involving pilot strategy in air-to-air combat, in an ECM environment with air-to-air combat, and as a model for fire control systems, weapon effectiveness studies, etc. It may be possible to use the model to conduct flight path and performance optimizations in which flight control system lags, damping, etc. significantly affect results. Variation of these parameters would then show system departure from the optimum results when responses and performance changes occur due to battle damage, weather effects, or equipment malfunctions. We do expect this type of a simulation to be valuable in preliminary design, prior to the need for a full system simulation.

VI. Summary

We have proposed a simulation model based on controlling directly the airplane response characteristics This model is currently being programmed and will be validated against in-flight simulation results. We expect this simulation model to be a valuable tool in many areas of research and development of flight control technology.

REFERENCES

1. Military Specifications, "Flying Qualities of Piloted Airplanes", MIL-F-8785C, November 1980.

2. Moorhouse, D.J., "The History and Future of Flying Qualities Specifications", AIAA Paper 79-0402, 17th Aerospace Sciences Meeting, January 1979.

3. Buckner, J.K., J.E. Walker III and C. K. Clark, "The Design of the F-16 High-Alpha Flight Control Characteristics and Control System Concept", AIAA Paper 79-0403, 17th

Aerospace Sciences Meeting, January 1979.

4. McRuer, D., Ashkenas, I., and Graham, D., "Aircraft Dynamics and Automatic Control", Princeton University Press, 1973.

5. Brulle, R.V., Moran, W.A., and Marsh, R.G., "Direct Sideforce Control Criteria for Dive Bombing", AFFDL-TR-76-78, September 1976.

6. Hodgkinson, J., and LaManna, W.J.., "Equivalent System Approaches to Handling Qualities Analysis and Design Problems of Augmented Aircraft", AIAA Paper 77-1122, August 1977.

7. Hodgkinson, J., "The Application of Equivalent Systems to MIL-F-8785B" in AFFDL-TR-78-171, December 1978.

8. Hodgkinson, J., and Snyder, R.C., "Flight Evaluation of Augmented Fighter Aircraft", AIAA Paper No. 80-1611, presented 17th Atmospheric Flight Mechanics Conference, August 1980.

9. Tryon, P.F., and Buckley, J.E., "Six Degree of Freedom Ground Based Simulation of Augmented Fighter Aircraft Longitudinal Characteristics", MDC A6914, February 1981.

10. Neal, T.P., and Smith, R.E., "An In-Flight Investigation to Develop Control System Design Criteria for Fighter Airplanes", AFFDL-TR-70-74, December 1970.

11. Smith, R. E., "Effects of Control System Dynamics on Fighter Approach and Landing Longitudinal Flying Qualities", AFFDL-TR-78-122, March 1978.

12. "LATHOS - Lateral-Directional High-Order System", NT-33 Flight Test Data, FDL Report in preparation by Calspan.

13. "Flying Qualities of Large Airplanes" TIFS - Flight Test Data, FDL Report in preparation by Calspan.

14. Cooper, G.E., and R.P. Harper, Jr., "The Use of Pilot Rating Scale in the Evaluation of Aircraft Handling Qualities", NASA TN D-5153, April 1969.

15. Hoh, R.H., et al, "Development of Handling Qualities Criteria for Aircraft with Independent Control of Six Degrees of Freedom", AFWAL-TR-81-3027, April 1981.

LOW-VISIBILITY VISUAL SIMULATION WITH REAL FOG

Wendell D. Chase*
Ames Research Center, NASA
Moffett Field, California

Abstract

An environmental fog simulation (EFS) attachment was developed to aid in the study of natural low-visibility visual cues and subsequently used to examine the realism effect upon the aircraft simulator visual scene. A review of the basic fog equations indicated that two major factors must be accounted for in the simulation of low visibility — one due to atmospheric attenuation and one due to veiling luminance. These factors are compared systematically by (1) comparing actual measurements to those computed from the fog equations, and (2) comparing runway-visual-range-related visual-scene contrast values with the calculated values. These values are also compared with the simulated equivalent equations and with contrast measurements obtained from a current electronic fog synthesizer to help identify areas in which improvements are needed. These differences in technique, the measured values, the features of both systems, a pilot opinion survey of the EFS fog, and improvements (by combining features of both systems) that are expected to significantly increase the potential as well as flexibility for producing a very high-fidelity low-visibility visual simulation are discussed.

Notation

B_{h} = ambient Sun luminance as observed toward horizon, cd/m^2

B_O = object-scene display luminance, cd/m^2

$\underline{B}_O$ = background display scene luminance, cd/m^2

B_R = object-scene luminance at pilot's eye position, cd/m^2

$\underline{B}_R$ = background-scene luminance at pilot's eye position, cd/m^2

C_R = apparent contrast

C_R/C_O = contrast transmittance ratio or contrast modulation

P = air pressure, kg/cm^2

R = horizontal range, m

RVR = runway visual range

R_V = horizontal fog visual range, m

t = time, sec

V = air velocity into environmental chamber, m/sec

Z = aircraft altitude, m

Z_P = vertical oscillator signal

Z_V = vertical oscillator signal from computer, V

Z_{VR} = vertical breakout altitude of aircraft relative to RVR, m

α' = display field of view through windscreen, deg

$\bar{\delta}_i$ = pilot control input variables to aircraft equations of motion, rad

θ_M = motor drive position of environmental chamber faceplate, deg

σ = extinction coefficient, 1/m

σ_1 = scattering coefficient, 1/m

σ_2 = absorption coefficient, 1/m

ω_Z = vertical altitude oscillator frequency, rad/sec

I. Introduction

When aircraft are landing or taking off, they are subject to strict Federal Aviation Administration (FAA) policies and procedures that govern the use of both the airspace and the airport runway facilities. Operations of aircraft that are allowed to fly through an approved airspace and that are granted certain landing privileges, are monitored by the FAA by means of a flight plan. The major reason for both pilot and FAA modification to these flight plans is weather. Weather causes delays, affects operational costs, and has a significant effect on flight safety. Because man has been virtually unable to alter the weather, constantly upgraded cockpit instruments and landing aids, such as the autoland and microwave landing system, are being used to permit pilots to land aircraft safely in spite of adverse weather conditions. Nevertheless, it is recognized that certain low-visibility weather conditions near airports can still contribute to unsafe landings. To help avoid such situations or to otherwise advise aircraft pilots, the FAA monitors visibility conditions during periods of rain, fog, or snow. Poorer visibility conditions are classified with regard to the approach requirements as Category 1, Category 2, or Category 3 (a, b, or c). Category 1 ILS approach allows the aircraft to descend to not less than 61 m (200 ft) decision-height altitude, with a transmissometer-measured runway visual range (RVR) of 731.5 m (2400 ft) or 804.7 m (2640 ft) observed visibility from the tower without operative touchdown zone and runway lights. This RVR may be lowered to 548.6 m (1800 ft) when the touchdown zone and runway lights are operational. A Category 2 ILS approach allows the aircraft to descend to a 30.5-m (100-ft) decision-height altitude and a transmissometer-measured RVR of 365.8 m (1200 ft) or a 402.3 m (1320 ft) observed visibility from the tower, with both touchdown zone and runway centerline lights operational. Category 3 (a,b,c),

*Research Scientist. Member AIAA.

without the aid of instruments (such as an autoland or microwave system), are blind landing conditions with visibility sufficient only for taxiing the aircraft along the surface of the runway.

Today, in spite of an FAA adverse weather monitoring system, advanced weather radar devices on board the aircraft, cockpit instrumentation, and ground aids (such as Visual Approach Slope Indicator, or VASI), as well as improved approach lighting systems for low-visibility final approaches, the critical problem remains: poor visibility is a major contributing factor in many terminal-area landing approach accidents. Recent National Transportation Safety Board (NTSB) accident statistics show that 48.3% of all air carrier accidents are caused by or related to adverse weather conditions.[1,2] According to another NTSB 10-year statistical analysis, 41% of all fatalities were caused by weather conditions.[3] A summary of 17 low-visibility accidents showed that 80% occurred when visibility was less than 1609 m (5278 ft) because of fog and rain, and 60% of these incidents occurred in nighttime conditions.[2] The underlying factor may be the result of faulty visual perception caused by distorted or reduced visual inputs occurring under conditions of rain and fog.[4]

An early attempt to use the natural effects of actual fog was made at the University of California at Berkeley (under the sponsorship of the FAA). Workers there constructed a large building (circa 1964) in which actual fog was produced and used in studies of airport lighting systems.[5-7] A 0.1-scale lighted runway was constructed and enclosed in a building approximately 245 m (804 ft) long. An overhead cable was raised about 8 m (26 ft) above the ground and a simple cockpit mockup was attached to the cable. Fog was generated throughout the building until 0.1-scaled RVR readings were at the correct value; then the cab commenced to travel (at about 1 m/sec) down the suspended cable. Observers inside the cab viewed the scene through the windshield as the cab descended through the fog toward the 0.1-scale lighted runway. The presence of actual fog served to increase the "realism" required for a balanced lighting system investigation under low-visibility conditions. Because actual fog was being used, conditions were present that could not be successfully reproduced by other methods. These were: 1) enveloping fog and its proximity enhanced the three-dimensional effect (although there was no way of measuring the accommodation reflex, this author believes that the observer's eyes accommodated to a nearby resting position for zero visibility rather than to infinity); 2) the fog was turbulent and not uniform; 3) the runway lights were attenuated correctly and a halo effect was produced normally; 4) a veiling luminance effect was present during the daytime caused by sunlight entering through various side windows along the building; and 5) scaling of the visibility ranges could be related to full-scale visibilities.

This simulation facility, which has since been dismantled, was very limited in its ability to: 1) provide the pilot with aircraft closed-loop controls, instruments, or motion other than the one open-loop motion constrained by the cable over which the suspended cab traveled through the fog; 2) control the fog to raise the visibilities rapidly (1-2 sec or less) for various types of breakout, as would be encountered when actually flying through broken clouds or patchy fog; 3) change the visibility with altitude as if encountering different vertical layered or gradient-type fogs; 4) simulate realistic speed and altitude (to do so would still require the construction of large and costly buildings); and 5) control the veiling luminance (caused by sunlight entering various side windows) — the fog was illuminated from the side rather than from the top.

The FAA has recently considered the benefits of upgrading and promoting the additional use of simulators to expand training and certification to improve safety, to increase fuel conservation, and to reduce training costs as well as airport congestion (according to an FAA-NPRM 14 CFR parts 61 and 121). A DOT/FAA Final Rule 121-14C effective August 29, 1980, declares three major phases for upgrading current simulators to permit and present realistic training in various abnormal and weather flight conditions that may be encountered during line operations. In regard to the Phase 2 visual-scene weather presentations, the requirements are for realistic representations of the following: 1) variable cloud density; 2) partial obscuration of ground-scenes-effect of scattered to broken cloud deck; 3) gradual breakout; 4) patchy fog; 5) the effect of fog on airport lighting; and 6) Category 2 and 3 weather conditions. The Phase 3 visual-scene presentations additionally include the sound, visual, and motion effects of entering light, medium, and heavy precipitation near a thunderstorm on takeoff, approach, and landings below an altitude of 610 m (2000 ft). Thus, implementation of visibility conditions of sufficient realism is required to fulfill both Phase 2 and Phase 3 training requirements.

What is not clear from the above requirements is how the low-visibility visual cues are to be improved, or what level of realism is required, or how the realism level is to be measured. Furthermore, there is confusion among those working in flight simulation about the properties of fog that are needed to accurately create simulation models of fog. As a result, the following basic discussion is intended to provide a needed basis from which a low-visibility model can be simulated and applied in the construction of simulation hardware devices.

II. Background

Types of Fog and Characteristics

Because the DOT/FAA Final Rule-121-14C mentioned above asked for realistic representations of clouds or fog, it is important at this point to differentiate briefly between a cloud and fog. A cloud is composed of aerosol droplets with a diameter of 20-50 μm and formed by adiabatic expansion (convective rise of warm air to higher altitudes.)[8,9] Fog is composed of smaller aerosol droplets with a diameter of 1-20 μm. Haze is composed of even smaller particles, with a diameter of 0.2-1 μm. Furthermore, a turbid atmosphere that permits a visual range of less than 1000 m (3280 ft) is considered fog, and lesser states of turbidity are referred to as haze.[10] The most common types of fog are: 1) radiation; 2) advection; and 3) frontal. Radiation fog is formed when the ground surface is colder than the air temperature; it can be further classified as "shallow" (single layered) or "mature" (deep multilayered). Advection fog is formed when

a warm moist air mass moves over a cold surface. Sea fogs are actually advection fogs. Frontal fog is formed as a boundary between a cold and warm air mass. Reference has sometimes been made to an "upslope" fog, but it is the early stage of a cloud that is forming as a result of convective adiabatic expansion.[9] Patchy fog could be formed from any one of the above three types because of local differences between the surface and air temperature. In cooler months, for example, ground depressions are known to have lower temperatures than the surrounding air; this temperature difference can contribute to the formation of random or patchy fog pockets.

It has been recognized for a long time that a wide divergence of opinion exists pertaining to the measurement techniques for recording and defining the optical viewing properties of fog.[11,12] To clarify some misrepresented facts concerning fog, Fig. 1 shows an object illuminated in bright daylight as it would normally be observed in clear air and in fog. (The edges are shown undiffused because the intense daylight veiling luminance is considered to mask the halo luminance and therefore could not be observed.) Figures 2 and 3 show an example nighttime scene taken at the Arcata (Calif.) Airport commencing from a light fog (Fig. 2) to heavy fog (Fig. 3). These photographs show 1) the presence of a halo that rapidly diminishes in heavy fog; and 2) a contraction of the apparent size of the halo light between light and heavy fog. A brief explanation of these effects is presented in Fig. 4, which shows how a distant light would be observed at night in the presence of a light fog to produce both an attenuation (loss of luminance) and the halo effect mentioned previously. Rays of light emanating from the point-source light are partially absorbed and diffracted by the fog droplets along the path to the observer. With each fog droplet encounter, the luminance of the light decreases, and at the same time the light spreads. The spreading of this light becomes the halo as seen by the observer.

The observed optical characteristics of fog are known to include the following factors: 1) attenuation of scene brightness, and 2) the veiling luminance effect from an ambient light source, such as from the sun or from aircraft landing lights. The veiling luminance is caused by light rays that may be multiply refracted and diffracted from one fog particle onto other particles to produce a scattering of the light as though it was emanating from all directions. Consequently, any scene to be viewed through the fog would not be visible until the contrast between the fog and scene improved above a certain contrast value.

Equations for Actual Fog

Atmospheric attenuation is a function of many variables: wavelength, path length, pressure, temperature, humidity, and the composition of the atmosphere.[13] The factor commonly used to describe the density of the fog is called the extinction coefficient. It is known to be composed of two parts: a scattering component (σ_1) and an absorption component (σ_2) so that (from Refs. 8, 13, 14)

$$\sigma = \sigma_1 + \sigma_2 \qquad (1)$$

The dominant term, σ_1, is that due to scattering from both the air molecules (Rayleigh scattering) and scattering by the aerosol particles (Mie scattering).[13] The average extinction coefficient for the visible spectrum (0.38-0.72 μm) at sea level depends as follows on the horizontal visibility range R_V:

$$\sigma = \frac{3.41}{R_V} \qquad (2)$$

(Koschmieder assumed a contrast threshold value of 2%; that value, which resulted in the present computation of $3.91/R_V = [1/\sigma \ln(1/0.02)]$ has tended to persist in meteorological circles, although it has been the subject of considerable doubt.[10] In another more recent study, Politch[15] found that the extinction coefficient should be $\sigma = 3.41/R_V$. This implies that the average value of the contrast threshold should be 3.3% as computed from $[(1/\sigma)\ln(1/0.033)] = 3.41/R_V$ which represents an average value for the visibility.)

This extinction coefficient is used to help formulate the luminance of an object seen in daylight and is composed of two parts: 1) attenuation; and 2) veiling luminance.[10,13]

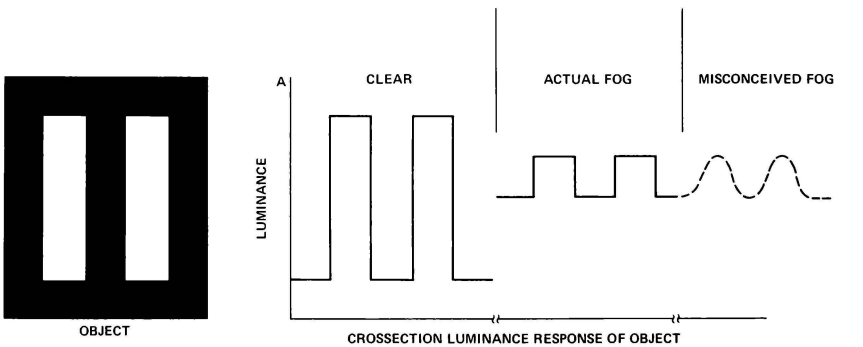


Fig. 1 Cross-section luminance characteristics of object illuminated in daylight for both clear air and fog.

Fig. 2 Effects of fog on runway lights — Arcata (Calif.) Airport.

Fig. 3 Effects of increasing fog on runway lights — Arcata (Calif.) Airport.

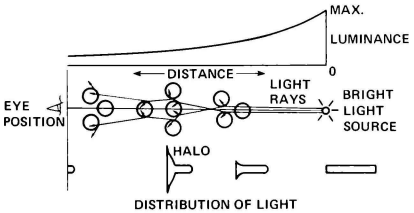
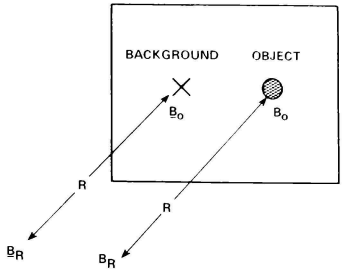


Fig. 4 Light attenuation in presence of fog.

$$\left.\begin{aligned} B_R &= B_o\, e^{-\sigma R} + B_h(1 - e^{-\sigma R}) \\ \underline{B}_R &= \underline{B}_o\, e^{-\sigma R} + B_h(1 - e^{-\sigma R}) \end{aligned}\right\} \qquad (3)$$



The attenuation of brightness difference:

$$B_R - \underline{B}_R = (B_o - \underline{B}_o)e^{-\sigma R} \qquad (4)$$

Inherent contrast:

$$C_o = \frac{B_o - \underline{B}_o}{\underline{B}_o} \qquad (5)$$

Apparent contrast:

$$C_R = \frac{(B_o - \underline{B}_o)e^{-\sigma R}}{\underline{B}_R} = \frac{C_o \underline{B}_o\, e^{-\sigma R}}{\underline{B}_R} \qquad (6)$$

Contrast transmittance or transference ratio:

$$\frac{C_R}{C_o} = \frac{\underline{B}_o\, e^{-\sigma R}}{\underline{B}_o\, e^{-\sigma R} + B_h(1 - e^{-\sigma R})} = \frac{1}{1 + \dfrac{B_h(1 - e^{-\sigma R})}{\underline{B}_o\, e^{-\sigma R}}} \qquad (7)$$

Contrast modulation transfer — where $\underline{B}_o$ = average scene luminance:

$$\frac{C_R}{C_o} = \frac{1}{1 + \dfrac{B_h(1 - e^{-\sigma R})}{\underline{B}_o\, e^{-\sigma R}}} \qquad (8)$$

The first term in Eq. (3) pertains to the luminance of the object and to its attenuation caused by the absorption and scattering coefficients of fog or cloud conditions. Upon examining Eq. (3) and assuming the range to the object R to be greater than the visibility range R_V, the first term (object luminance) would approach zero and the second term (caused by veiling) would approach a maximum value equal to B_h. Of course, if the range were zero in the first term of Eq. (3) — the observers eye touching the object — there would be no attenuation because there would be no fog droplets to see through. Similarly, for the second part of Eq. (3), and at a zero range condition, the veiling effect would be zero. Furthermore, if the fog density or extinction coefficient approaches zero irrespective of the range (R) — which means there are no fog droplets or air molecules present in the atmosphere — then the object would not be attenuated and the veiling luminance would be zero. Thus,

Eq. (3) satisfies the most obvious initial conditions. The main reason it is so difficult for a daytime observer to distinguish an object is that the sunlight falls on a cloud or fog and produces an intense scattering of light. The veiling luminance of the fog so greatly exceeds the reduced luminance of the object scene that it makes it difficult if not impossible to see. The method used to better describe how well an object can be seen can be determined by examining the contrast transmittance ratio or contrast modulation (C_R/C_o) shown in Eqs. (7) and (8).

Equations for Electronic Fog

An early electronic method of fog simulation used a flying spot scanner, raster-driven in parallel with a model-board television camera.[16] A movable transport film located in front of the scanner, with a film density proportional to the fog, transmitted the light through the film to a photocell and generated a proportional signal that was mixed with the television scene video signal. Another widely used technique in current use is actually a refined version of the above method in which the flying spot scanner and the variable density film have been eliminated. An example of a modified raster system of this type now in use at Ames Research Center operates by switching the proportional fog and background scene video inputs through a gain-changing amplifier before it terminates at the display monitor.[17] Circuitry to maintain pitch and roll is synchronized to overlay the fog with the horizon and ground view scene presented on the final display scene monitor. Although details of the following material are beyond the scope of this report, a brief discussion is justified in order to demonstrate the simplified fog and contrast equations which are currently represented by electronic methods and which can be directly compared with the actual fog equations presented earlier. In referring to Fig. 5, a runway is projected onto the display monitor CRT. Above the screen vertical sweep (Z_{sweep}) position, corresponding to the visibility R_V, a constant washout signal is added. To generate a smooth contrast transition between the sky and ground scene objects so that the fog does not appear as a sharp line, the amount of fog washout signal added is smoothly reduced down to zero at the object scene screen sweep position (Z'_{sweep}) on the display, corresponding to 0.1 R_V. The simulated fog effect is variable over the simulated region of width:

$$\Delta R = R_V - 0.1\,R_V \tag{9}$$

The signal added simulates the veiling luminance contribution and can be represented in terms of a fog luminance equation as

$$B_R = B_o + B_h \qquad \text{for} \quad R \geq R_V \tag{10}$$

$$B_R = B_o + B_h\left[1 - 0.1\,\frac{R_V}{R}\right] \qquad \text{for} \quad 0.1R_V \leq R < R_V \tag{11a}$$

$$B_R = B_o \qquad \text{for} \quad R < 0.1R_V \tag{11b}$$

For the purpose of this paper, Eq. (11a) represents an area of interest and is used to represent the equation between an object and its background as follows:

Equation of an object seen in simulated daylight:

$$B_R = B_o + B_h\left[1 - 0.1\,\frac{R_V}{R}\right]$$

$$\underline{B}_R = \underline{B}_o + B_h\left[1 - 0.1\,\frac{R_V}{R}\right]$$

Attenuation brightness difference:

$$B_R - \underline{B}_R = (B_o - \underline{B}_o) \tag{12}$$

Inherent contrast:

$$C_o = \frac{B_o - \underline{B}_o}{\underline{B}_o} \tag{13}$$

Apparent contrast:

$$C_R = \frac{B_o - \underline{B}_o}{\underline{B}_R} = \frac{C_o\underline{B}_o}{\underline{B}_R} \tag{14}$$

Contrast transmittance or transference ratio:

$$\frac{C_R}{C_o} = \frac{\underline{B}_o}{\underline{B}_o + B_h[1 - 0.1(R_V/R)]} \tag{15}$$

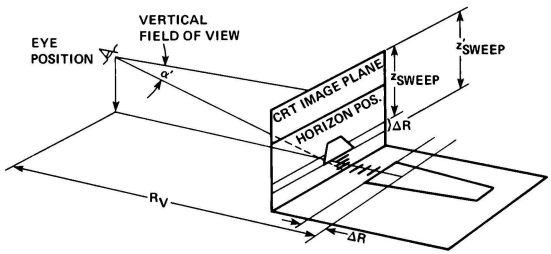


Fig. 5 Projection of scene onto CRT plane and parameters used to control visibility by electronic methods.

Contrast modulation transfer — where $\underline{B}_o$ = average scene luminance:

$$\frac{C_R}{C_o} = \frac{1}{1 + \frac{B_h[1 - 0.1(R_V/R)]}{\underline{B}_o}} \qquad (16)$$

Variations of the above method are in use with current computer-image-generated (CIG) displays that also include raster-driven displays. Some CIG displays may use algorithms to change visibility by modifying each picture element of the display; however, they may be limited in a real-time generation of the low-visibility scene.

Perceptual Limitations of Electronic Fog

Although electronic contrast adjustment could be done according to the correct contrast equations, it would still have limitations. The synthesized veiling luminance caused by the Sun, Moon, or landing lights is in the optical plane of the CRT, which is usually collimated near infinity. The pilot should be less subject to the Mandelbaum effect — the accommodation to a nearby resting distance in the absence of strong distant cues.[18] Also, contrast adjustment of the electronic fog does not simulate the halo effects that are prominent at night. Furthermore, the fog is two-dimensional and is homogeneous.

Because these differences in the present electronic fog method presented conflicting low-visibility visual cues compared with the natural cues of actual fog, it seemed reasonable to attempt to construct a simulation device that would use actual fog. It was also recognized that this new device might also have limitations and might require some elements of the electronic method for precisely producing a high-fidelity low-visibility simulation. In order to examine the low visibility environment in greater detail, an apparatus from which low-visibility measurements are obtained for comparison with the actual fog equations is described, and the modifications needed for reproducing a high-fidelity low-visibility visual simulation are discussed.

II. Environmental-Effects Fog Chamber

Description

Many of the conventional simulators currently in use are constructed with top-mounted display CRT's viewed through mirror beam splitters and a spherical reflective mirror positioned in the simulator windshield. The empty space between the beam splitter and windshield was a likely area in which a small environmental (fog and rain) chamber, through which the pilot could see while viewing the out-the-window scene, could be positioned. Although the above optical system could be designed to accept an environmental or fog and rain attachment, it was convenient, for experimental purposes, to use a previously successful optical system, which uses either a color television model-board or computer-image-generated (CIG) scene monitor located at the focal plane of the collimating lens (Fig. 6). In this arrangement, the empty space between the beam splitter and the windshield seemed an appropriate

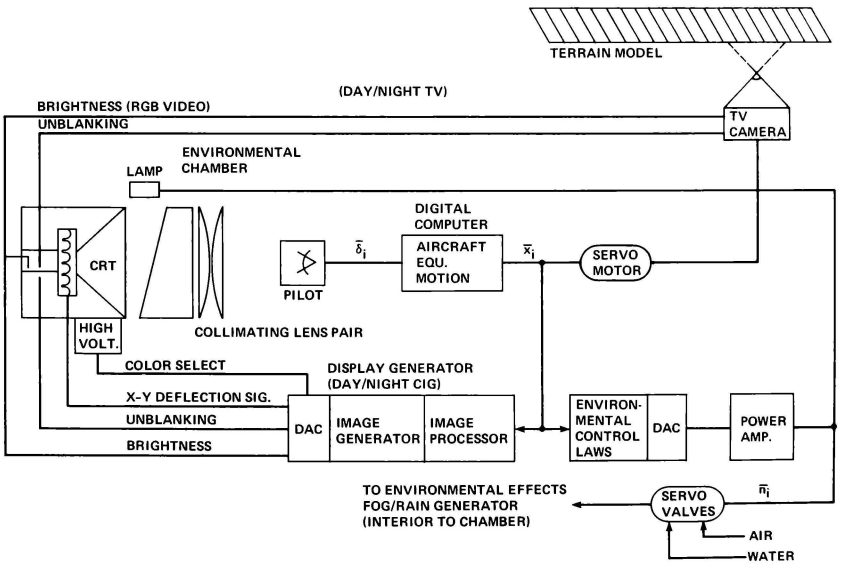


Fig. 6 Piloted closed-loop environmental effects visual display system.

place to experimentally position a small environmental fog and rain chamber through which the pilot could see the collimated display scene and use it in performing the necessary closed-loop final approach procedures and maneuvers under low-visibility conditions. Consequently, a program for developing "An Atmospheric Low-Visibility Environmental Visual Display Attachment for Aircraft Simulators" was undertaken.[19] The subject device was conceived and developed — within the Man-Vehicle Systems Research Division, Ames Research Center — for use in conducting research on the low-visibility scene. The device (patent pending — NAS-ARC-11158-1) is capable of providing fog, rain, or both fog and rain combined. For the purposes of this report, however, only the details of the low-visibility, fog-generating system are presented.

In referring to Fig. 6, the components needed to support and test the operation of the new environmental attachment are 1) a main-frame digital computer; 2) a display generator; 3) a color display monitor, such as a beam-penetration type or a color television monitor; 4) collimating lens arrangement; 5) the environmental chamber and fog generators; and 6) an ambient light source. For the purposes of evaluation, the small digital computer within the display generator was used to provide the longitudinal aircraft dynamics and control laws to the chamber. The arrangement of the above components is shown in Fig. 6. The computerized display from the computer-image-generated-scene monitor is observed through both the collimating lens pair and the environmental chamber. Interior to the chamber, at the sides, are two primary environmental effects fog generators (not shown). Normally, the two primary fog generators would suffice; additional units would be used only when very high recovery is necessary, such as when breaking in and out of broken clouds. Positioned between the top of the environmental chamber and the face of the display monitor is a fluorescent lamp for simulating the ambient veiling luminance; the brightness of the lamp is controlled by the digital computer.

Figure 7 is a photograph of the experimental hardware developed for installation in the windshield area of the aircraft simulator cab. The two large, 63.5-cm (25-in.) diameter collimating lenses are combined to achieve short focal length; they are usually mounted in the aircraft simulator windshield. They are used to form a virtual image from the picture presented upon the 53.3-cm (21-in.) display monitor (located at the focal plane of the lens). The resultant virtual image is viewed by the pilot, at a distance from the windshield of about 63.5 cm (25 in.), through the fog generated within the environmental chamber. An example of what the pilot would see is shown in Figs. 8 and 9. Figure 8 is a runway view looking through the chamber (without fog) of a TV model-board visual scene; Figure 9 is the same scene observed through a 305-m (1000-ft) visibility (RVR). Although the above discussion pertains to only one aircraft windshield window, it should be emphasized that multiple environmental chambers can be positioned at each window of the

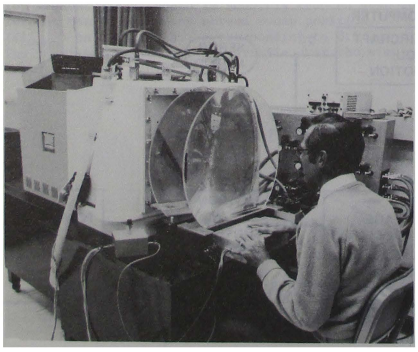


Fig. 7 Arrangement of experimental component hardware.

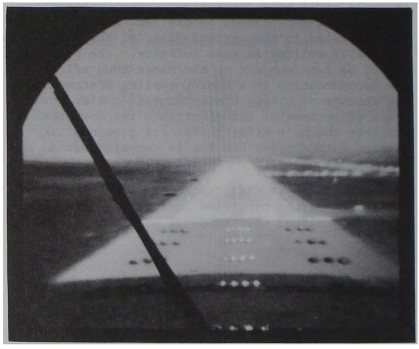


Fig. 8 Virtual display observed through collimating lens pair and chamber without fog.

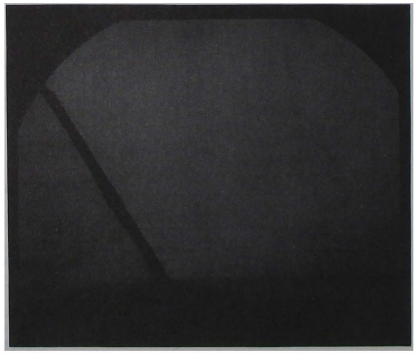


Fig. 9 Display scene observed through collimating lens pair and chamber with fog calibrated to 305-m (1,000-ft) RVR.

simulator cab, each of which could operate independently with varying visibilities, if so desired.

Method of Operation

The device that produces the natural low-visibility environment is discussed in more detail in Ref. 19. However, it is this device that manufactures the fog (which is composed of minute aerosol droplets). The droplets are ejected at a low velocity — about 0.3 m/sec — into the environmental chamber. As the number of these fog droplets within the chamber increases, they become more tightly packed within the constant volume of the environmental chamber, changing the attenuation coefficient $\sigma \cdot R$. Consequently, the maximum density is reached as the attenuation coefficient approaches 20; at that point, it becomes impossible for an observer to see any object through the small central thickness (0.24 m) of the environmental chamber. A single 12-W overhead fluorescent lamp produces an effect that can make the fog appear even more dense, when it is properly controlled; the lamp is used to simulate the effect of sunlight falling upon the fog or cloud.

The remaining major problem was how to remove portions of the entrapped fog in a manner representative, to the pilot, of realistic conditions, such as a low-ceiling breakout or a gradual day or night low-visibility fade-in of the runway.

Contrary to popular belief, very moist air is lighter than dry air. When dry air is introduced into the environmental chamber it is heavier and more readily displaces the lighter, moist or fog-laden air. The desired effect can be more naturally achieved by allowing dry air to enter the bottom of the environmental chamber and at the same time allowing the excess saturated or lighter moist air to escape through a pressure relief valve located at the top of the environmental chamber. Mixing of the dry air and light moist air also produces the effect of a reduction in density, and hence the extinction coefficient also changes accordingly.

The velocity of the dry unsaturated air entering the environmental chamber is set approximately equivalent to the aircraft altitude rate-of-descent. This appears to produce the right turbulence effect as the moist, foggy air is driven out the top of the environmental chamber.

Air entering the environmental chamber, if not interrupted in some manner, would cause all the fog droplets to be evacuated from the environmental chamber in a very short time. For some conditions, such as simulating a low ceiling breakout, this effect may be desirable. However, for the most part, it would also be very desirable to attempt to control the volume of air entering and leaving the environmental chamber in such a manner as to simulate accurately the entire range of visibility conditions that the pilot usually encounters.

The details of the primary mechanism that can precisely remove and replace portions of air from the environmental chamber are discussed in Ref. 19. Briefly, however, assume the chamber is filled with fog and is at the maximum density ($\sigma \cdot R = 20$). A computer-generated variable-altitude oscillator signal ω_Z produces an output signal Z_P which is converted to an analog voltage signal Z_V. This oscillating voltage signal commences to repeatedly energize an external relay whereupon dry air at pressure P and moving at velocity V is admitted in pulses to the bottom of the environmental chamber to mix with the fog. As a result, both the dry air and fog are forced to exit at the top via pressure relief valves. The frequency of the altitude oscillator, ω_Z, changes as a function of altitude and RVR in order to produce digital pulses based on the following equation:

$$\omega_Z = \frac{30[Z - (Z_{VR} - 15.24)]}{30.48} \tag{17}$$

where Z_{VR} is a function of RVR. The oscillator output signal Z_P produces voltage pulses Z_V through a digital-to-analog converter ranging between 0 and 10 V; the voltage pulses are frequency dependent:

$$Z_P = (1 - \cos \omega_Z t) \tag{18}$$

Since $Z_P = Z_V/5$ V

$$Z_V = 5(1 - \cos \omega_Z t) \tag{19}$$

Thus, the shift in frequency is then made to correspond to a specific number of air pulses relative to the RVR condition. This method of calibration shows that the clean dry air entering the chamber in proportional air pulses displaces a compact volume of fog within the chamber, resulting in calibrated changes in visibility.

Improvements in Fog Simulation Fidelity

Earlier, in Eq. (3) two factors were shown to be present in a daylight scene: 1) that due to the object and its attenuation through the fog; and 2) that due to the ambient light upon the fog, referred to previously as the veiling luminance. A closer examination and measurement of these two principal characteristics is now possible under the controlled conditions of the environmental fog chamber. Thus, in order to determine the degree of validity for the simulated low-visibility environment, the individual contributions of attenuation and veiling luminance was measured at the pilot's eye position. To examine the pure veiling luminance contribution only, a Pritchard photometer was positioned at the pilot's eye position to record the luminance values (obtained by reducing B_o to zero in Eq. (3) with the use of a black velvet cloth equal in area to the CRT monitor scene and positioned at the collimating lens focal plane) while the visibility was made to change. The precision injection of calibrated pulsed air into the chamber causes the fog density to change — changing the extinction coefficient — and the ensuing visibility change is then compared with the veiling luminance term in Eq. (3). The calibrated RVR values recorded by the photometer (with a veiling light luminance of 17.15 cd/m^2 (5 fL)) while looking through both the optical center of the collimating lens and environmental chamber for RVR's ranging between 0 and 3,660 m (12,000 ft) is shown in Fig. 10a. This measurement compares and demonstrates both analytically and empirically that as the fog becomes more dense, the ensuing brightness increases to a maximum value at zero visibility. Also, it should be noted that no correction for altitude has been added so

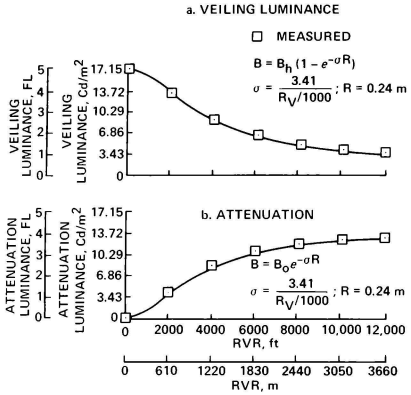


Fig. 10 Changes in veiling luminance and attenuation with RVR: a) veiling luminance; b) attenuation.

this effect is what would be expected along a horizontal path at sea level and in a shallow radiation fog along the ground.[13] This veiling luminance can also be expected to change, somewhat, depending on other types of fog and by considering the choice of an altitude-related extinction coefficient correction (discussed in more detail in Refs. 13 and 19).

The other factor is that caused by attenuation. To obtain an attenuation response through the fog, a 17.15-cd/m² (5-fL) fiber optic point-source light (2-mm diameter) was positioned at the collimating lens focal plane and the ambient veiling lamp was turned off. Again, air was pulsed into the environmental chamber under computer control to change the visibility. The photometer was used to record the attenuation values, which are presented in Fig. 10b and in Fig. 4. This demonstrates that a decrease in brightness occurs with a reduction in simulated RVR. To illustrate characteristic properties of actual fog that are present, Fig. 11 shows a sequence of six photographs of a fiber optic point-source light recorded with increasing fog density. Of particular interest are the similarities, as shown in Figs. 2-4 for three phenomena: 1) a halo becomes more prominent with light fog and disappears for heavy fog; 2) an apparent contraction of the size of the halo light occurs as the fog becomes more dense; and 3) the edge sharpness of the light inside the halo appears to be well defined. The halo may actually be expanding, but the change in luminance, as the fog becomes more dense, may be below the limit of perception and hence not readily visible.

The above method provides many advantages: 1) piloted closed-loop control of the aircraft with the visual sensation of flying through actual turbulent fog; 2) ability to control the turbulent fog as if raising or lowering the visibilities rapidly for various types of fog, including breakout or broken-cloud effects; 3) scaling of the fog within the small environmental chamber (this has made large and costly fog research buildings obsolete); 4) a realistic veiling luminance effect that changes with RVR;

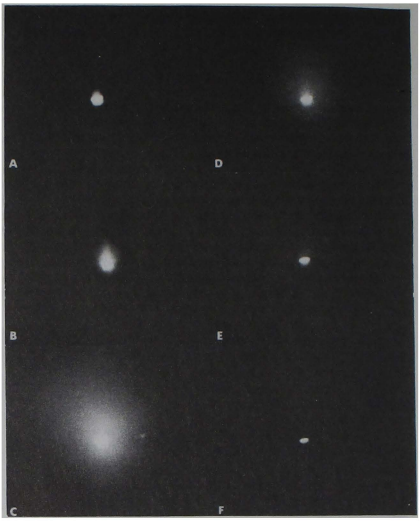


Fig. 11 Sequence of a single point-source light attenuated by increasing fog density: a) no fog; b) very light fog; c) light fog; d) medium fog; e) heavy fog; f) dense fog.

5) a realistic point-source light attenuation effect; 6) control over the veiling luminance which provides realistic contrast thresholds; 7) allowing the pilot's eyes to accommodate according to the phenomenon of empty-field myopia to occur as it may in real-world featureless scenes; and 8) the nearby turbulent fog provides a three-dimensional effect rather than a two-dimensional uniform fog produced and observed in the optical plane at infinity.

Fidelity Limitations

The environmental chamber was initially designed for use in investigating 1) the physical characteristics of fog; 2) a method for generating fog and introducing it within the chamber; 3) a control law to rapidly change visibilities within the chamber; 4) an anticipated blending with electronic methods to partially veil the background in order to improve the fidelity; and 5) a means for ultimately producing combinations of both fog and rain. Consequently, the environmental chamber was originally constructed with a fixed-position faceplate (Fig. 6). The author believed also that this configuration could be used to simulate a mature radiation type of fog with the use of a fixed faceplate position with the support of some electronic veiling of the background. It can be shown that the extinction coefficient for this type of fog increases with altitude, and therefore the fog becomes much more dense with altitude.[19] Thus, the visibility that the pilot may encounter may very well be practically

zero for some altitudes. Furthermore, another reason for providing an initial zero visibility in the simulator was to insure that no scene elements could be observed by the pilot (as the aircraft descended through the fog) until objects on the ground were within the RVR. Although the veiling effect is correct, two correctible factors occur: 1) there is no gradual fog gradient so the fog appears too dense in the foreground area, but not unlike a patchy ground fog; and 2) the daylight scene elements appearing in the background are seen too clearly for the stated RVR condition. Even if the faceplate was designed for some other fixed position, the small thickness of the chamber could not provide an overall fog gradient to completely obscure objects in the scene beyond the encountered RVR or visibility condition.

Fidelity Improvements with Electronic Techniques

The two situations discussed above can be resolved by combining a basic feature already present in the electronic method. It should be pointed out that the electronic method was shown — from examination of the fog equation (3) and contrast equation (7), and comparison with the electronic equation (11) and contrast equation (15) — to be incapable of producing a realistic fog. However, by utilizing the vertical sweep signal as it relates to RVR (previously shown in Eq. (9)), a harmonious screening or veiling of the background scene used in conjunction with the actual intervening fog can be used to create a more realistic three-dimensional fog depth. Furthermore, because an intervening fog is actually present, the overall veiling luminance of the sky will be technically more correct for all visibilities. Therefore, in the present study, an attempt was made to provide a limited background screening effect for the daylight scenes equal to approximately 6-km (3.7-miles) visibility which was held constant simply because current hardware constraints prevented a complete synchronization.

IV. Performance of Environmental Fog Chamber

Pilot Evaluations

To obtain preliminary information on the effectiveness of the low-visibility simulation, six airline pilots participated in a fixed-base simulation study. All pilots were on current flight status and qualified in similar type aircraft. A DC-8 jet transport aircraft was simulated with dynamics that included only the longitudinal dynamics and autopilot. This was because 1) the speed and small size of the resident PDP-11/05 Display Generator Computer limited the number of real-time calculations, and 2) the flights were made without the usual instrument assistance, thereby forcing the pilot to establish dependence on the vertical and longitudinal out-the-window visual cues within the display scene. A side arm controller and an autopilot disengaging button were incorporated into the simulation. The hardware components (see Fig. 6) provide 1) the background display scene; 2) the aircraft response to the pilot's controlled input through the aircraft equations of motion; and 3) the controlled operation of the intervening fog effects and visual scene conditions which were synchronized by the computer.

The evaluation used three test conditions. These (Table 1) were 1) a static daylight photographic scene of Moffett Field runway 32L; 2) a color television model-board dynamic scene; and 3) a color computer-image-generated (CIG) dynamic nighttime scene of the San Jose (California) Municipal Airport. Standardized initial conditions for all flights were as follows: an altitude of 269 m (883 ft) above the runway in a zero RVR visibility condition; 4.86-km (3-mile) ground distance to the runway threshold; an approach airspeed of 135 knots; landing gear down, and a flight path of -3° from the horizontal. No flight performance measures were obtained for the simulated landings and pilot monitoring observations since this test was largely exploratory. A pilot opinion survey, pertaining to the fidelity of the visual simulation, was used to help evaluate the display and to help isolate deficiencies where necessary.

Table 1 Test conditions

Ambient light environment	RVR visibility, m (ft)	Display scene
Day	Maximum visibility 3,218 (10,558) 1,609 (5,279) 805 (2,641) 402 (1,319) 61 (200) (low ceiling)	Static (Moffett Field)
Day	Maximum visibility 3,218 (10,558) 1,609 (5,279) 805 (2,641) 402 (1,319) 61 (200) (low ceiling)	Dynamic (TV)
Night	Maximum visibility 3,218 (10,558) 1,609 (5,279) 805 (2,641) 402 (1,319) 61 (200) (low ceiling)	Dynamic (CIG)

Results of Pilot Evaluations

The instructions to the pilots and the questionnaire will be presented in a separate report. The test conditions shown in Table 1 were presented to the pilots who rated the display conditions for fidelity on a scale of one (high fidelity) to four (low fidelity) at the end of each flight session. Only one 2-hr session was required for the pilot to complete the test sequence and to record his answers.

The pilot's cursory responses relative to the three display conditions presented in Table 1 are summarized in Fig. 12. For convenience and comparison, the results of a similar fidelity survey, which was conducted in the previously described University of California/FAA fog facility, are also presented in Fig. 12. In assigning a level of realism over the range of visibilities for the static daylight display, a mean value of 2.17 and a standard deviation of 0.84 were found. This is similar to that established for the FAA fog facility — the latter had a rating of 1.7 and a standard deviation of 0.53. Two common complaints by the pilots about the static daylight scene, which may have contributed to the slight difference in mean value, were 1) there were no approach lights in operation so that they could not be seen first through the fog; and 2) the fog had an appearance of being patchy, but not entirely

unrealistic, in the vicinity of the approach segment in front of the runway threshold. This type of fog looked more like that of a ground fog or that for a low ceiling breakout.

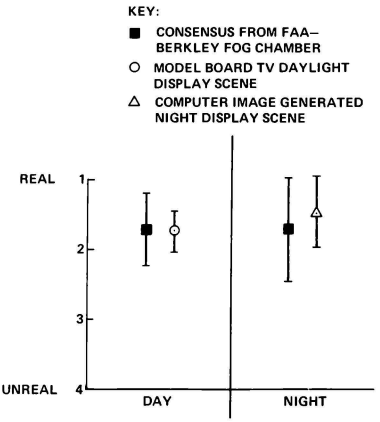


Fig. 12 Pilot opinions of low-visibility visual simulation fidelity compared with that of the real world.

For the dynamic televised daylight scene, the mean level of realism was 1.83 with a standard deviation of 0.37. This was nearly equivalent to the FAA fog facility fidelity, with more agreement among the pilots, as evidenced by the much smaller standard deviation. The pilots attributed more realism to this type of display because they could see the approach lights moving through the fog in a manner to which they were accustomed, although they did complain about the poor resolution of the typical television display. Figure 12 includes also the nighttime fidelity ratings obtained from the FAA fog facility; they show a mean of 1.65 and a standard deviation of 0.53. No differences were noted in the ratings obtained from the computer-generated (CIG) nighttime display, which had a mean fidelity of 1.5 and a standard deviation of 0.5. The CIG standard deviations and a symmetric distribution showed that the pilots were equally divided, half of them believing that this nighttime low-visibility simulation was as real as they had experienced. This increase in low-visibility realism, the author believes, may be partially attributed to the intensity modulation of the distant lights relative to the RVR and to their appearance as they emerged through the intervening fog.

Contrast Measurement

Another measure of the effect of the display scene on the low-visibility simulation is provided by the previously developed contrast transmittance equations for both actual fog (Eqs. (3)-(8)) and electronic fog (Eqs. (11)-(16)). These equations can be used to predict the transmitted contrast of a scene through either the actual fog or electronic fog. A comparison of the results could then be used to differentiate between the two methods for simulating low visibility. Furthermore, the contrast measurements are more valid relative to human psychophysics in which a 2% value is regarded as the contrast threshold (the World Meteorological Organization recommends a 5% value). This author prefers to use 3.3% for the reasons discussed previously pertaining to Eq. (2).

To measure the contrast transmittance, the TV runway scene shown in Fig. 8 was used. A schematic of the display scene is shown in Fig. 13. The position designated as B_o was actually the left-most touchdown zone hash mark at the runway threshold, and $\underline{B}_o$ designated a position just outside the runway apron. Thus, the measurements were taken between B_o and $\underline{B}_o$ through the actual fog and at the same positions for the electronic fog for visibilities (RVR) ranging between 305 and 3,650 m (1,000 and 11,975 ft). These results and the predicted values calculated from the contrast transmittance equations for actual fog (Eq. (7)) and electronic fog (Eq. (15)) are presented in Fig. 14. It can be seen that the measured values are very close to the predicted values and that the fog chamber appears to produce a valid contrast. The electronic fog is shown to be unrealistic because the contrast 1) is much lower than it should be throughout the visibility spectrum; 2) is shaped differently across the entire RVR spectrum compared with the actual fog; and 3) is in violation of the accepted values for limits of human contrast perception. It is possible that this can be corrected, but only by completely redesigning the hardware in compliance with the actual fog equations and comparing contrast measurement results for validation.

Suggested Improvements

The underlying pilot comments pertaining to the low-visibility display scene and those that appeared to divide the fidelity ratings were 1) that the static display scene elements all appeared as if at the same RVR; 2) the dynamic television displays appeared to have too much of the scene visible for the stated RVR; and 3) that for the nighttime display, the emerging lights through the fog relative to the RVR conditions appeared too bright. The above three factors derive from the absence of a fog gradient, which becomes more prominent when close to the ground; all of them are related, in part, because

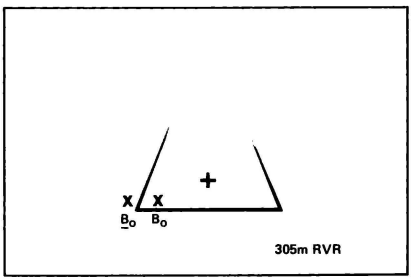


Fig. 13 Simplified schematic of display scene.

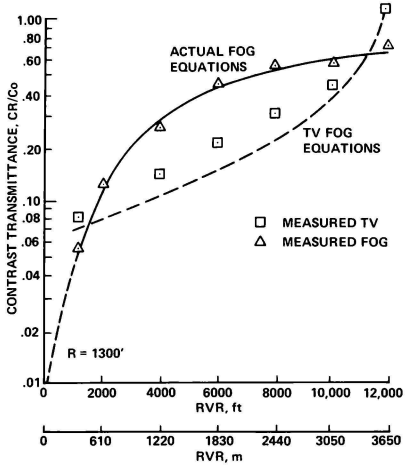
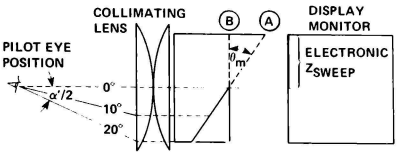


Fig. 14 Contrast transmittance for actual fog and electronic fog as a function of visibility.

of the original design of the environmental chamber and, in part, because the display itself is two dimensional. It was anticipated that the environmental chamber would eventually operate with a synchronized fog RVR and electronic vertical sweep to appropriately obscure the background. However, for this study, the vertical sweep was held constant because of hardware design limitations. Although correctable, the above hardware change is not sufficient to allow a natural appearing fog gradient for visibility situations including three-dimensional objects encountered near the ground. Figure 15 shows an improved environmental chamber designed to accommodate a fog gradient through the use of a movable faceplate, which changes positions according to altitude and RVR. When in position B, there would be a maximum fog in the chamber until the aircraft altitude Z reached a predetermined RVR-derived vertical breakout altitude Z_{VR}. At that altitude, the faceplate would begin to rotate to position A, according to the motor drive (θ_m) equation shown in Fig. 15. As the aircraft descended, the observed composite scene would not only have the appearance of a realistic fog gradient, but would also have a background obscured in the right proportions to the RVR. This modification has been incorporated in a new chamber; preliminary observations are very favorable, although more tests will be needed to determine precisely the overall effectiveness.

An artifact, due to the design of current CRT display monitors, was also responsible for making the more distant nighttime runway lights (as they came into view) appear too bright through the fog. This problem has been recognized, and a means for adding an intensity modulation correction term for each light source relative to the observed RVR, in addition to the improvements discussed above, would



WHERE: $Z_{VR} = R_V \sin(\alpha'/2)$

AND: $\theta_m = 45 \left[1 - \frac{Z}{Z_{VR}}\right] \quad 0 > \theta_m < 45$

Fig. 15 Faceplate position and motor drive equation for helping to provide a natural fog gradient.

perhaps increase the level of realism above that which has been currently obtained.

V. Summary

Currently, there is an emphasis on conducting all pilot training in flight simulators, principally because of 1) the high costs of actual aircraft training flights and certification; 2) the need for reducing airport congestion and improving air safety. Associated with this emphasis is the demand for more realism in the visual simulation display environment. Visual simulation is now advancing rapidly, especially with the new technique of constructing computer-generated images. The low-visibility physics and the methods for synthesizing the environmental or meteorological conditions (as in the past and during the development of those displays) have not been well understood nor coordinated. As a result, the simulated environment has been aesthetically created and calibrated without standards to create unnatural visibility conditions. It is not known whether this technique will remain in operation in lieu of the new FAA simulator ruling. Furthermore, because the physical atmosphere or the atmosphere dynamics have not been present or included in the aesthetically adjusted landing displays, the validity and level of low-visibility realism using current methods is highly questionable. Therefore, this author conceived that a small environmental chamber containing actual fog particles could be constructed within the space between the display monitor and the windshield-positioned collimating lens. It was felt that this technique would allow further exploration and understanding of the physics of low-visibility atmospheric conditions as well as providing a means for increasing the validity of the visual scene contrast effects as perceived by the pilot. Consequently, a device was constructed that 1) produces actual fog; 2) includes a environmental chamber to entrap the fog, resulting in RVR values from "clear" down to "zero-zero"; 3) includes an RVR control system that has been calibrated and found to be accurate within 2% of the theoretical atmospheric values; and 4) allows the pilot to make unconstrained, closed-loop trajectory, final approach or takeoff maneuvers under any day or night visibility condition.

To further explore a new technique for synthesizing a more realistic low-visibility environment and to make constructive modification to the equip-

ment by documenting both favorable and adverse potential user comments, a preliminary study using six senior airline pilots was conducted. Three display conditions with five basic variations of visibility were presented: 1) a static daylight photographic scene of Moffett Field runway 32L; 2) a dynamic daylight television scene obtained from a model-board-type visual attachment; and 3) a nighttime dynamic CIG color scene of the San Jose (California) Municipal Airport. The results of this cursory study showed that the above three display conditions for both day and night were a significant improvement over current methods and that they were very realistic, except for several factors which could be corrected with minor changes in the hardware. The adverse comments indicated that an improvement in representing a fog gradient would be desirable.

The pilots believed that the realism effect with the presence of actual fog enhanced the displays; they considered the computer-image-generated (CIG) nighttime display to be nearly the equal of their real experience. The pilots' fidelity or realism ratings for both the daytime and nighttime series of low-visibility conditions, by comparison, were found to be equivalent to the ratings taken from the FAA fog facility, which also used actual fog. Contrast measurements of the display scene observed through actual fog were in very high agreement with the theoretical values. Therefore, the subject low-visibility environmental chamber attachment appears to have the potential for reproducing a wide range of realistic visibility conditions and other stressful distractions (such as a nearby lightning flash) as well as for providing an increased flexibility to conduct terminal-area piloted flight maneuvers under other adverse environmental conditions. Further high fidelity improvement can now be obtained by including a variable-position faceplate and by combining the scene with some of the electronic fog synthesizing methods (now in use) to enhance the fog gradient and contrast perception needed to provide a better three-dimensional effect. These results support the hypothesis that in the simulation of low visibility, the presence of actual fog enhances the perception of the visual scene cues in a manner that portrays more realism.

References

[1]"Annual Review of Aircraft Accident Data, U.S. Air Carrier Operations," NTSB-ARC-78-2, 1977.

[2]"Flightcrew Coordination Procedures In Air Carrier Instrument Landing Approach Accidents," NTSB-AAS-76-5.

[3]"Ten Year Accident Statistics," SB-81/3021, Sept. 22, 1980.

[4]"A Review of Pilot Performance As Related to Visibility During Adverse Weather," Texas Transportation Institute, Texas A & M University, College Station, Texas, July 1976.

[5]Haines, R. J., "Effects of Various Runway Lighting Parameters Upon The Relation Between Runway Visual Range and Visual Range of Centerline and Edge Lights In Fog," Department of Transportation Report No. FAA-RD-73-170, Dec. 1973.

[6]Finch, D. M., Horonjeff, R., Paula, H. G., and Ahlborn, G., "An Investigation of Intensities for The U.S. National Standard Runway Touchdown Zone and Centerline Lighting," Interim Report No. 5, The Institute of Transportation and Traffic Engineering, U. of California, Berkeley, Jan. 1964.

[7]Finch, D. M., Horonjeff, R., and Paula, H. G., "Evaluation of Runway Lighting Systems for Effectiveness In Dense Fog," Final Report RD-65-58, The Institute of Transportation and Traffic Engineering, U. of California, Berkeley, Jan. 1966.

[8]Middleton, W. E., Vision Through the Atmosphere, University of Toronto Press, 1963.

[9]Hettel, H. J., dePena, Rosa G., and Pena, Jorge A., "Controlled Generation of Large Volumes of Atmospheric Clouds in a Ground-Based Environmental Chamber," NASA TM X-3266, 1975.

[10]Tricker, R. A. R., Introduction to Meteorological Optics, American Elsevier Publishing Co., New York, 1970.

[11]Horonjeff, R., "Requirements for Runway Lighting," Shell Aviation News, Issue No. 331, 1966.

[12]Beck, R. H., "The Hostile Environment of Low Visibility," presented at the 15th ALPA Air Safety Forum, Seattle, Wash., July 9-11, 1968; also in the Air Line Pilot, Nov. 1968.

[13]RCA Electro-Optics Handbook, EOH-11.

[14]Koschmieder, H., "The Theory of Horizontal Visual Range," NASA Technical Translation, NASA TT-F-14,930, June 1973.

[15]Politch, J., "Contribution of Particles Pollution to Visibility," Optical Engineering (SPIE), Vol. 16, No. 1, Jan./Feb. 1977.

[16]Coen, R. T., "Electronic System for Varying Fog Simulation with Changes in Direction of Sight," U.S. Patent 3,524,019, Aug. 11, 1970.

[17]Moots, E. E. and Chase, M., "Electronic Fog Generator for Flight Simulation," NASA CR-152265, 1978.

COMPUTER GENERATED IMAGE SYSTEM TRENDS FOR THE 21ST CENTURY

81-0983

R.S. Rulon
Engineering Specialist
Grumman Aerospace Corporation, Bethpage, NY

Abstract

A computer generated image research and development program is being conducted to develop a system to meet the simulation requirements for training for the next generation of aircraft. In planning the program, system design and performance goals were determined after investigating vehicle types and flight profiles expected to be employed in the next twenty years. The effect of future sensor systems which require coordination between sensor and visual environment simulation was investigated. System design goals were formulated after review of LSI and mass memory state-of-the-art. The resulting program planning and design goals are discussed.

Introduction

About 1976 the Grumman Company embarked on a Computer Image Generation System Development Program. It was recognized that if such a program were to be a success, long range goals would have to be adapted, since the development cycle would be lengthy, followed by a rather lengthy amortization period. In essence, this means that the system or its derivatives would be in operation well into the 21st century. Also, there is little justification for another "me too" system only matching the performance of existing systems that have proven their adequacy in many training applications and are being continuously upgraded in performance. What was needed was a careful assessment of requirements for the next generation of simulators prior to planning the program.

It has become trite to speak of the rapidity of technological advancement, making the prognostication of future visual system performance and design a risky business. In reality, many trends can be forecast without resorting to the circumlocution of the oracle of Delphi if we examine factors which are presently discernible that will affect the area of concern. In the case of visual systems, these parameters are both external and internal to the training environment.

The external factors are defined as those derived from the introduction of new vehicles, greater reliance on improved sensors and the development of new tactics in military operations. The internal factors are defined as those hardware and software advances in the state-of-the-art, which in turn, provide the possibility of increasing CIG performance while containing systems total costs. These increases in simulator capability will of themselves expand the use of CIG as tasks formerly not performed in the simulator or levels of proficiency training previously unobtainable in simulators become a reality.

External Factors

The present, nearly a ten year lag from planning to introduction of new weapons systems, helps us to anticipate the types of vehicles we will find in the early 21st century. We can expect a mix of high performance jets and helicopters and, despite less than feverish interest, we will find VSTOL and STOL aircraft in both military and cargo versions.

The environment in which these aircraft will operate is increasingly more hostile. Increased deployment of battlefield missiles and sophistication of electronic warfare has already placed a premium on low level and nap of the earth operations for survival. If first day losses after start of hostilities are to be kept at an acceptable level, more than trained pilots are required. Tactics must be developed and practiced under such conditions of dynamic realism that correct defensive the offensive maneuver decision making approaches the speed of reflex actions.

The acquisition and recognition of possible launch sites, missile launch and missile trajectories under all environmental conditions must be simulated with considerable accuracy to develop the tactics and skills that can be transferred to the real world situation. Tactics to combat new aircraft and weapons suites adapted by a future advisory must similarly be refined and practiced.

Tactics will be influenced by the advanced sensor systems to be found on future aircraft. The dependency upon these sensor systems and their accuracy will require that future visual systems displays are closely coordinated with the output of the simulated sensor systems. In the past, sensor/visual display coordination has usually been limited to rather gross coordination of altitude cues for altimeters, radar video and instrument landing systems. In future systems, both sensor images and sensor derived range and altitude readouts will have to correlate within very small percentages.

Further VSTOL and STOL vehicles will be expected to operate from surfaces such as heaving small ship decks, jungle clearings, or hilly or even mountainous terrain with little or no landing aids, causing the pilot to rely very heavily upon visual reference.

Fly by wire, sophisticated sensors and flight computers will unburden the pilot of many control decisions; however, these same systems will encourage precise flying, close to terrain or covering obstacles. This will not only increase the need for visual/sensor coordination but will also increase the need for accurate and subtle visual wing and rotor clearance cues, since judgement of range and depth perception will be of great importance in this type of air work.

Visual depth perception is often considered in terms of stereo vision. While not totally discounting the value of stereo vision at ranges beyond wing tip or rotor clearance, scene detail rapidly becomes more important than stereopsis for distance measurement. This is very demanding on visual simulation, since the pilot must rely upon identification of familiar objects and features. Such subtle cues as

grass, leaves or dust blown about by the vehicles down wash are of possible importance. Unfortunately, reliance on feature identification for altitude discrimination can be difficult, as in many parts of the world the features of terrain look very much alike from different altitudes. In simpler terms, little rocks at low altitude look like big rocks at higher altitude. Simulating the difference in the appearance of terrain as an effect of altitude will therefore be very demanding on resolution and image content.

Tactics will also require acquisition and recognition of targets or missiles at the greatest possible range, further stressing the need for high resolution. The combat environment with hostile high performance fighters and missiles of many types will require wide fields of view for tactics development and practice. Wide fields of view will also be required for nap of the earth maneuvering as peripheral vision becomes important in attitude discrimination. For fighter aircraft, not only will the full visibility envelope of the aircraft be required in the simulator, but any mirrors in the cockpit will also be required if simulator capability is to be fully exploited for tactics development and proficiency training.

The external factors affecting visual display requirements are the same as those we have experienced from the beginning of visual simulation. Safely extrapolating from these immediate trends we can prophesy that we will face still more demands for more scene detail, more resolution and wide fields of view. There will be a different mix of requirements for each training or simulation application so the ideal system is one which can be tailored to provide different performance for different usage in the most cost effective manner.

Internal Factors

The factors internal to CIG technology are those advances in components and subsystems which in many instances are already discernible. The nature of much of the processing in visual display systems is that much high speed bit manipulation involved with moving, storing and ordering of binary information is required. Advances in high speed Random Access Memories increase the capabilities possible in future systems and reopen for investigation algorithmic techniques not exploited in the past, due in whole or in part to hardware limitations. LSI and VLSI technologies, coupled with the availability of custom and semi-custom chips, have given systems designers a new flexibility in implementation concepts. This provides the developer the opportunity to design systems architectures less costly to implement through reduced manufacturing cost and debug time.

The desire for increased simulation capability to meet next generation requirements has previously been discussed. A safe generalization has always been that increased performance implies increased complexity. Increased complexity has always brought with it increased life cycle cost, as procurement costs have risen due to more circuitry and components to buy, design, manufacture, integrate and test and higher maintenance costs in the field. The promise of VLSI and LSI to enable more logical structuring and compartmentation of functions for troubleshooting and modularized repair should have a beneficial cost advantage in systems maintenance costs.

The architecture envisioned using the present.. state-of-the-art, with growth capability to take advantage of coming improvements, should utilize much parallel processing to save computational time. Parallel processing is utilized in present systems; however, as the VLSI technology expands we can expect new architectures to exploit further the advantages of increased savings in computation time by use of this technique. How this advance is utilized is a trade off between reducing lag times by speeding up total throughput time or increasing CRT update rates or increasing scene content.

The choice of display systems will remain dependent upon the application of the host simulator. That is, landing and take off operational flight trainers will remain on the low end of the image content and field of view requirement, while full mission, air combat simulators will represent the high side of these requirements. However, another generalization is that the user will wish the widest fields of view he can justify. As in the past, we can expect a continual slow, steady improvement in the resolution of color CRT displays and increased choice of high brightness projection systems. Advanced dynamic electronic image distortion correction will free us from some of our most vexing optics problems, while CRTs demonstrating very high orders of image stability will encourage use of multiple off-axis projectors for wide angle systems requiring edge matching of CRT rasters.

As resolution, scene content, fields of view and dynamic performance have increased in the past, we have seen how flight training simulators have moved from part task training to full mission training and even into tactics development. Pressure for greater simulator usage will continue as cost savings, safety and training flexibility continue to become more attractive, resulting in expanded simulator capability demands. More emphasis can be expected in simulator usage for maximizing crew proficiency until we approach the Star Wars concept to move from simulator to operational performance. Although it may sound futuristic, it did happen in space flight almost twenty years ago and continues to this day.

Program Goals

The pressures which are discernible on future flight simulator development thus established, a CIG development program can be structured. Having established future simulation usage requirements and future equipment state-of-the-art advances in the gross sense, the detail program goals must be determined in a systems approach considering the inter-relationships of image generation algorithms, image generation hardware, data base generation, storage and retrieval, image transmission and image display. As in any large scale system, tradeoffs between idealizing one parameter or subsection and impacting other subsections must be made throughout the program. Thus, the development program becomes an iterative process of parallel investigation, followed by systems reviews, trade off decisions and re-investigation as required. The program initiated by Grumman was originally designated Multi-Micro Image Generation System (MMIGS) as the acronym

neatly described the system architecture as originally conceived. After some period of time, this name was revised to Advanced Computer Image Generation System as more descriptive of total program goals. The program was thus designed to encompass four major interdependent elements:

- Algorithm Development
- Image Generation System Development
- Data Base Generation Techniques
- Display Techniques.

Of the four elements, display techniques are the least controllable by the CIG systems researcher as being dictated by the particular type of simulator being considered and the state-of-the-art of proprietary equipment.

To begin investigation, the nature of the displayed image must be reviewed. In determining image requirements for simulation, much has been made of "adequacy for training", as it should be due to ever present cost constraints. In developing a new system concept, adequacy for training becomes a rather vague concept as precise systems utilization is not established. The guide then becomes flexibility to accommodate a range of systems performances which can be achieved by modular design concepts, the upper performance bound being a system capable of performance providing realism not now obtained for tasks not now fully performed to a lower bound of systems adequate for part task training.

Image content for nap of the earth maneuvering can be taken as a worst case for terrain simulation, though not necessarily incorporating all special effects such as damage assessment, aerial perspective (atmospheric attenuation with range) and flight above clouds. From examination of NOE image requirements, the need to develop high levels of detail such as individual trees, shadows, cultural objects and terrain texture was developed. The need for such cues as tree leaves moving, grass blowing and dust kick up was also determined, as was the requirement for moving models, with shadows fixed to the ground surface.

Low level flight, while theoretically requiring a lower level of detail due to faster flight, added the requirement for texture over relatively large areas and accurate terrain contour representation. Ground attack and air combat maneuvering added damage assessment and moving aerial targets such as missiles and hostile aircraft. Low level high speed flight and aerial combat in high performance aircraft placed emphasis on viewpoint maneuverability, high update rate and reduction in transport delay. Thus, while many simulators would not require the total "worst case" CIG image capability, any future system must provide the capability to cover the performance spectrum.

A shopping list of image content and enhancements was generated which included:

- Terrain contour accurately generated from a data base correlated with radar, IR low level TV or laser ranging systems simulation in the host device
- Cultural features such as buildings, runways, roads and lights
- Transluscent shadows related to sun angle to provide cues of object size, relation to ground plane and to aid in target detection
- Clouds, including cloud top, cloud bottom, flight through solid cloud and scud
- Moving models, including air and surface vehicles with their attendant shadows
- Spatially fixed terrain texture related to topography, vegetation, geology or cultural feature of an area
- Dynamic texturing for generation of special effects, such as leaf and grass motion due to rotor down wash
- Water, including varying sea states
- Atmospheric effects such as attenuation due to fog, haze or smoke as a function of range and density and aerial perspective (change in color toward blue gray with range)
- Glare and glint dependent upon surface reflectivity and sun angle
- Reduction in the distracting effects of spatial and temperal aliasing inherent in raster based CIG systems
- High image update rates together with reduction in system lag from command input to image display.

The desire to increase resolution and to reduce aliasing effects is an important parameter in developing a preliminary system concept before beginning the development of the image generation algorithms. The preliminary system architecture was developed to partition key computational tasks and to provide a straw man system for refinement through data flow analysis to evaluate real time system feasibility.

The development of algorithms begins at the high end of the scale, that is the capability to generate a scene which will be a good analogy of the real world with a high level of scene content. The scene must contain atmospheric effects, and have provisions for generation of moving models time of day and other special effects. Algorithms would also have to be generated to compensate for limitations of raster CRT based display systems that result in distracting artifacts within the image.

Edge based algorithms have proven themselves in many years of CIG system usage and will continue to be employed. However, attainment of curved features, which comprise the major content of real world terrain, and many man-made objects, use large numbers of edges requiring large data bases that are costly to generate. To model the subtle nonlinearities of the real world, a more efficient approach using primitives more suitable for generation of curved surfaces to which shading and conformal texture is to be added must be investigated. The bicubic patch technique and the quadric surface technique have both been investigated as

possible candidates. The relative mathematical complexity of the bicubic patch technique relative to quadrics rules in favor of the latter as the most cost effective solution to curved surface generation. As the algorithms are developed, non-real time image testing must be performed both to evaluate the algorithm approach and also to refine the approach by determining the applicability of simplifications and approximations that would reduce overall system computational load without degrading display accuracy or image quality for simulator application.

The high level of detail to provide more subtle distance and velocity cues is simply too expensive to be built up by the same primitives used for basic terrain contours and important scene cultural elements. Texturing is already being employed as a practical solution to breakup areas in the scene that would otherwise appear in unnatural monchrome showing no change in appearance from different ranges or aspects. The simplest example of texture would be an orchard where rows of trees result in subtle changes in shading, giving a corrugated appearance to what would otherwise be a flat green field. In most cases, texturing will be still more subtle and irregular than in neat, geometric, tree plantations. Gently undulating terrain might present the toughest problem for use of texture to provide the very subtle tonal changes which are found in nature.

Once we accept that texture by adding a variable shading to colors within the represented image areas is a computationally efficient method, the use of texture to simulate more than surface feature nonlinearity comes to mind. Provision of cloud top, cloud bottom, broken cloud and cloud shadow by texture on terrain are usages of texturing techniques worthy of investigation, as the possibility of providing great detail and subtlety of real word features from a small data base is most appealing. In addition to the use of spatially fixed texture, more dynamic texture can be used to supply the moving dust and grass and even shaking leaves on a tree.

As the scene content within the CIG system becomes more closely related to the real world, the pressure to present the color tones within the image more realistically can be expected to increase. The acceptance of saturated colors, that is, the natural selection of the home TV watcher as the result of low contrast ratios within CRT displays, will be replaced by a desire for more natural colors, particularly if the simulator is to include ground attack and damage assessment tasks requiring target acquisition. Future high brightness high contrast display can be expected to provide the improved color performance which will surely be demanded. Thus, tonal descripters in the data base and tonal computations must consider the future demand for more realistic colors. Compounding the tonal computation problem will be the requirement that sensor image generation can be expected to be required in some simulators with the desire to use the same equipment and data base for visual image generation. The capability to process the additional data base descripters to simulate the differences in sensor physics from human vision will have to be accommodated. IR simulation must consider the thermal characteristics of the material, location of heat sources, time of day and atmospheric effects. If the tonal computation algorithms and data base design allows for this dual image capability at the outset, an efficient solution to the dual problem of visual and sensor system image generation is achievable.

The friendly and hostile moving models within the combat environment must be simulated using the same primitives as for terrain features. The moving models exhibit several unique requirements. They move about an axis system independent of the simulated aircraft or the earth axis system with velocity independent of motion about other axis systems. Land vehicle simulation faces the further problem in that the moving model must remain on the surface with its attitude determined by the surface on which it lies. Pressure to develop an efficient moving model position and orientation solution will also come from the users of land vehicle simulators such as armored vehicles. The accuracy of the solution is affected directly by the terrain contour generation algorithm and any simplifications or truncations employed in the computations for terrain.

Resolution, scene content and data base generation and management are examples of interrelated parameters when attempting to design an optimized system. Increased scene content is of little value if resolution is not adequate or if the data base becomes too costly to generate nor is resolution increase of much value without gain in scene content. Increased resolution, if it is to be exploited, requires managing data brought into the image computations such that computation is not wasted on unresolvable data while special consideration is given to objects of importance which would be resolvable by the human eye in the real world yet may be below the resolution of the display system. This indicates that as the program develops, image testing and system optimization must consider the interaction of such parameters.

The data base for future simulators must take advantage of the Defense Mapping Agency (DMA) Data Base generated for the world. Further, the ideal of automated data base translation, allowing new areas or changes to existing areas to be entered into the visual system data base at minimum cost and with very short lead time must be considered during system investigation. Data base generation must also allow generation and updating of small areas in the data base from other than DMA data, such as engineering drawings or reconnaissance photos and data. Once this rapid data base generation and modification capability is obtained, the ability to pre-fly a mission on short notice becomes a reality.

Program Structure

Having formulated generalized goals from observed trends, the program was structured, by time phases, to provide the logical progression of investigation, development, test and review necessary to contain cost, minimize risk and utilize research talent to best advantage. The phases shown in Fig. 1 are those recognizable as common to any R and D program. Further breakdown by unique CIG tasks are shown. Each task is shown flowing through a near identical cycle (vertical column) with the task approached in time essentially from left to right, with each task iterating through as many cycles as required to reach a feasible solution. Image testing is employed throughout both to test feasibility and to provide the basis for simplification. Preliminary image testing, performed in non-real time, is adequate for the early stages of algorithm development, but can be misleading. Simplifications or truncations which appear acceptable in non-real time can result in very distracting discontinuities

and instability from frame to frame in real time operation. Psuedo real time image testing is employed through the use of time lapse technique, allowing the scene to be viewed under dynamic conditions for final evaluation.

Analysis of computational loading is performed during algorithm selection and refinement and is also used for development of a more detailed systems architecture. The systems architecture is developed for the now state-of-the-art with the capability to best take advantage of what are seen as future trends. The architecture of the future will be impacted by the desire to exploit VLSI technology and high speed mass memory advances. The system will also be impacted by the need to accommodate future improved high resolution displays. The advantage of breaking up complex scenes into sections for parallel processing of a segment of the display raster offers an advantage in being easily adaptable to displays of higher resolution as additional parallel paths can be added to accommodate increased computation using the identical components.

The output of the first time phase is a set of refined algorithms which have been proven feasible by analysis and image testing, together with a candidate system architecturally configured to meet the anticipated computational size and timing requirements.

The second phase of the program is recognizable as a design breadboarding phase in which program risk is contained through build up of a portion of the system prior to build up of a full scale prototype. Final optimization of the algorithms also takes place at this time. At this time, it is possible to contain program costs for hardware development by taking advantage of the parallel processing architecture employed by the candidate system. As the image is processed in subsections, it is possible to design and construct one or more of the parallel computational paths for testing results which will still provide a very real test of the design. Figure 2 shows both phase 2 and phase 3.

The first step in this phase is to utilize the algorithms optimized previously and the conceptual device design to perform verification using a simulation of the VLSI machine design. This also will be an iterative process in which the compatibility of the algorithm and the machine design will be tested and both algorithms and hardware optimized. The optimization effort continues until computational assets are efficiently utilized. Overloading, which causes bottlenecks in some portions of the system under operating conditions, will be eliminated. Having achieved this verification together with optimization of the machine and the algorithms, the design and building of the device breadboard can take place using off the shelf integrated circuit components when practical. Testing and evaluation can take place using only a portion of the total image simulation, as previously stated.

PHASE II
ALGORITHM OPTIMIZATION
MACHINE SIMULATION VIA VLSI LOGIC SIMULATION
ALGORITHM/MACHINE COMPATABILITY VERIFICATION
ALGORITHM/MACHINE REFINEMENT
PERFORMANCE VERIFICATION
BREADBOARD DEVELOPMENT USING OFF THE SHELF COMPONENTS
TEST & EVALUATION
CUSTOM CHIP DESIGN
PARTIAL SYSTEM BUILDUP
TEST & EVALUATION
REWORK
FINAL TEST AND EVALUATION
PHASE III
PROTO TYPE ONE WINDOW SYSTEM DESIGN
SYSTEM BUILDUP
SYSTEM INTEGRATION
SYSTEM TEST & EVALUATION

Fig. 2 Phase II Breadboard System Phase III Prototype System

The third phase results in the final development of a prototype system. This phase will be the most expensive in both material and labor; however it will be of low risk due to the testing performed in phase 2. The prototype system will have full up capability with the only limitation being that only one display need be driven. The parallel nature of much of the computation plus loading analysis of common computional elements will be adequate to prove multiwindow capability. After completion of the prototype, final refinements can be made and performance optimized under realistic operating conditions.

	IMAGE PRIMITIVES	ANTIALIASING	TEXTURING	VISIBILITY LOGIC	SPECIAL EFFECTS	SYSTEM DESIGN
CANDIDATE CONCEPTS						CANDIDATE ARCHICTECTURE
ALGORITHM DEVELOPMENT						PERF REQMT ESTIMATE
NON REAL TIME IMAGE TESTING						
REFINE & SIMPLIFY						DESIGN CONCEPT UPDATE
COMPUTATIONAL ESTIMATE						ARCH. REFINEMENT
PSUEDO REAL TIME IMAGE TESTING						
VERIFICATION & REFINEMENT						SYSTEM DESIGN UPDATE

Fig. 1 Phase One Concept Development & Feasibility Investigation

81-0985

A CRITIQUE OF THE GRAVITY VECTOR ALIGNMENT METHOD FOR MOTION SIMULATION

Howard Jaslow *
Gould Inc.
Simulation Systems Division
Melville, New York

Abstract

Conventional simulators tilt the motion platform to align the gravity vector with the resultant specific force vector of the aircraft. But, false cues can still arise with this gravity vector alignment method. In fact, this method provides real cues only for limited maneuvers. In this paper, a new theory of tilt and motion perception is reviewed. Equations defining the current gravity vector alignment approach and its modification with the new approach are derived. A technical explanation is given of inherent errors and limitations of the conventional approach, along with a quantitative comparison with the new approach.

Background

It is common practice [1-4] with simulator motion drive algorithms to tilt the motion platform so as to align the gravity vector acting on the simulator with the resultant specific force vector of the aircraft. That is, the force of gravity in the simulation exercise is used to simulate a steady aircraft acceleration.

Quoting from Parrish [1], "the representation of sustained longitudinal force cues through rotational tilt to align the gravity vector is almost standard practice in motion simulation." This method stems from the work of Schmidt and Conrad [2,3] where they developed the motion drive equations. Quoting from their work [2], "... the specific forces resulting from motion of a simulated aircraft should be presented to the pilot in the cab so that it has the same direction with respect to the cab as the true specific force would have with respect to the cockpit of the aircraft."

In this method, it is tacitly assumed that as long as both the simulated and actual aircraft force vectors make the same angle with respect to the aircraft, then the sensation experienced by the trainee is the same as in the aircraft. This is assumed regardless of the magnitude of the aircraft acceleration. It will be shown below that this is not a valid assumption.

New Theory of Tilt and Motion Perception

A new theory of tilt and motion perception [5-7] and the experimental data [8] supporting this theory [5] clearly demonstrate that false cues can still arise with the gravity vector alignment method. This theory is based on the anatomical mechanism for sensing sustained specific forces as experienced in a constant acceleration state or when tilted in a gravitational field, or a combination of both. This physical mechanism in the inner ear, called the utricular otolith, is basically a platform which which can only be displaced parallel to its surface. Acceleration normal to the otolith platform is not sensed.

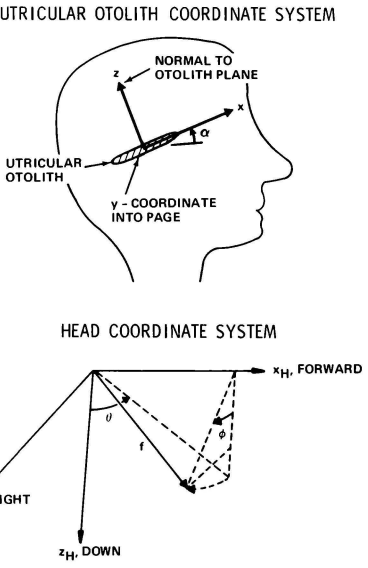


Figure 1. Coordinate Systems

The x and y components of the specific force (gravity vector minus acceleration vector) acting on the otolith in the otolith coordinate system (Figure 1) are as follows.

$$f_x = f(\cos\alpha \sin\theta - \sin\alpha \cos\theta \cos\phi) \quad (1)$$

$$f_y = f \cos\theta \sin\phi \quad (2)$$

where

* Manager, Research and Development
Member AIAA

f = specific force magnitude (number of g's)

f_x, f_y = components of specific force along otolith coordinates

Θ, Φ = pitch and roll angles of specific force with respect to head coordinate system (see Figure 1), respectively

α = geometric tilt back angle of otolith $\simeq 30^\circ$ (for more accurate results use $\alpha = 28.7^\circ$)

The f_z component is not required since it acts normal to the otolith plane and is not sensed.

Several conclusions can be drawn from these equations.

(i) Each acceleration input to the otolith is uniquely defined by two values: f_x and f_y.

(ii) Therefore, all linear acceleration inputs can be mapped onto a "sensation plane" having the coordinates (f_x, f_y).

(iii) Any two linear acceleration inputs having the same value of (f_x, f_y) are sensed as being one and the same motion.

(iv) f_x and f_y are two variables which are functions of three independent variables. This implies that there indeed exist combinations of pitch, roll and number of g's which give rise to the same motion sensation.

(v) The value of (f_x, f_y) provides a definite numerical value to be used in quantifying tilt perception.

Equations (1) and (2) can be used to relate pitch angle to a combined pitch and acceleration experience. As shown in reference 5, for pitch only ($\Phi = 0$), Eq. (1) yields

$$\text{pitch and acceleration: } f_x = f \sin(\Theta - \alpha) \quad (3)$$

$$\text{perceived pitch: } f_x = \sin(\Theta_p - \alpha) \quad (4)$$

For the same input, f_x, to the otolith sensor coming from a combined pitch and acceleration experience (Eq. 3) and from a normal experience in a 1-g field (Eq. 4 with $f = 1$), there results for the perceived pitch angle

$$\Theta_p = \sin^{-1}\left[f \sin(\Theta - \alpha)\right] + \alpha \quad (5)$$

This equation is plotted in Figure 2 for $\alpha = 28.7^\circ$ where it can be seen how the specific force, f, influences the perceived pitch angle. For example, a pitch of 15° with a specific force of 2 g's results in a perceived pitch of zero. Note also that all curves pass through $\Theta = 28.7^\circ$ because a specific force acting at this head pitch down angle is normal to the otolith and any change in specific force acting normal to the otolith is not sensed.

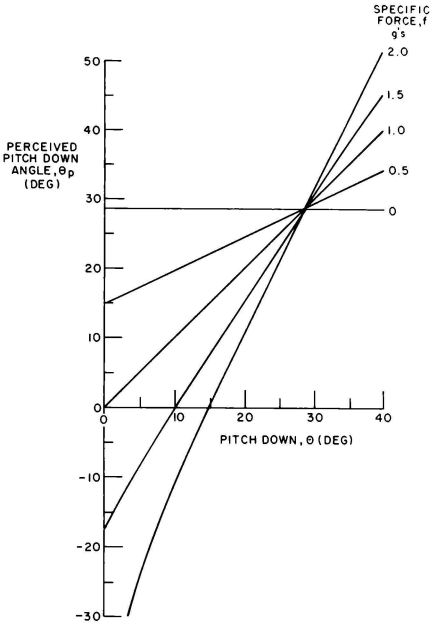


Figure 2. Effect of Specific Force on Perceived Pitch Angle

Equations (1) and (2) express the otolith components of the specific force vector $\bar{f}$ in terms of the magnitude (f) and orientation of $\bar{f}$ with respect to the head (Θ,Φ). For simulation, we need to express f_x and f_y in terms of the aircraft and simulator motion input. These are given as follows[6] with accelerations in g-units.

$$f_x = -\Big[\cos(\alpha+\delta)\sin\Theta_a + \sin(\alpha+\delta)\cos\Theta_a \cdot\cos\phi_a\Big] - \Big[a_{x_a}\cos(\alpha+\delta) - a_{z_a}\sin(\alpha+\delta)\Big] - \Big[(\dot{q}R_z - \dot{r}R_y)\cos(\alpha+\delta) - (\dot{p}R_y - \dot{q}R_x)\cdot\sin(\alpha+\delta)\Big] - \Big[\{p(qR_y + rR_z) - R_x(q^2 + r^2)\}\cos(\alpha+\delta) - \{r(pR_x + qR_y) - R_z(p^2 + q^2)\}\sin(\alpha+\delta)\Big] \quad (6)$$

$$f_y = \cos\Theta_a \sin\phi_a - a_{y_a} - \Big[\dot{r}R_x - \dot{p}R_z\Big] - \Big[q(pR_x + rR_z) - R_y(p^2 + r^2)\Big] \quad (7)$$

where

a_{x_a}, a_{y_a}, a_{z_a} = aircraft translational acceleration components along aircraft body axes $= \dot{u} + qw - rv$, $\dot{v} + ru - pw$, $\dot{w} + pv - qu$, respectively

u, v, w, = aircraft translational velocity components along aircraft body axes

R_x, R_y, R_z = distance vector components (from aircraft CG to head CG) along aircraft body axes

Θ_a, ϕ_a = aircraft pitch and roll angles, respectively

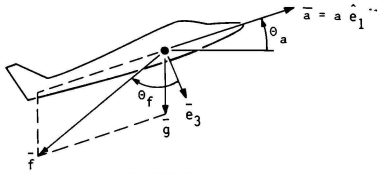
= seat tiltback angle

Note that the f_z component is not required since it does not contribute to sensation.

For simulation with real cues, f_x and f_y must be the same in the simulator as they are in the aircraft. Therefore, equations (6) and (7) are used as the basis for critiquing and modifying the gravity vector alignment method.

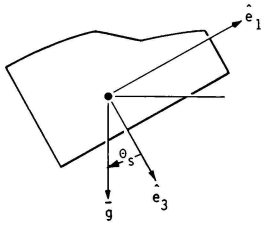
Conventional Gravity Vector Alignment Method

The conventional gravity vector alignment method directs the line of force acting through the trainee in the simulator to be the same as he would actually experience in the aircraft. This appears to be a logical approach for simulating sustained acceleration; and it would be correct if the magnitude of the forces were the same in both cases. But, this is not necessarily so. The false cues introduced by this method are discussed in the next section.



AIRCRAFT

GRAVITY VECTOR ALIGNMENT METHOD: $\Theta_s = \Theta_f$



SIMULATOR

Figure 3. Gravity Vector Alignment Method

To assure that the lines of force in the simulator and the aircraft are the same, the current practice is to tilt the simulator such that the angle subtended by the body axis with respect to the gravity vector is the same as that in the aircraft between the body axis and the specific force vector. This method is illustrated in Figure 3 for longitudinal motion with aircraft acceleration along the body x-axis. For this case, the angle Θ_f, between the aircraft z-axis and the specific force vector is derived as follows.

The specific force vector is given by

$$\bar{f} = \bar{g} - \bar{a} \quad (8)$$

where $\bar{g}$ = gravitational vector

$\bar{a}$ = aircraft acceleration vector.

From Figure 3,

$$\bar{g} = -g\sin\Theta_a\,\hat{e}_1 + g\cos\Theta_a\,\hat{e}_3 \quad (9)$$

$$\bar{a} = a\,\hat{e}_1 \quad (10)$$

where θ_a = aircraft pitch angle. Therefore,

$$\bar{f} = -(a + g \sin \theta_a)\, \hat{e}_1 + g \cos \theta_a\, \hat{e}_3 \qquad (11)$$

where $\hat{e}_1$ and $\hat{e}_3$ are the unit vectors for the body x and z axes, respectively. The angle, θ_f, between the body z-axis and the specific force vector is thus given by

$$\tan \theta_f = \frac{\bar{f} \cdot (-\hat{e}_1)}{\bar{f} \cdot \hat{e}_3}$$

$$\therefore \tan \theta_f = \frac{n + \sin \theta_a}{\cos \theta_a} \qquad (12)$$

where n = a/g = number of g's.

For the same line of action on the simulator, we impose the condition that the simulator tilt angle, θ_s, is equal to θ_f.

$$\therefore \tan \theta_s = \frac{n + \sin \theta_a}{\cos \theta_a} \qquad (13)$$

θ_s is the gravity vector alignment angle or simply the tilt angle of the simulator (see Figure 3). This equation is plotted in Figure 4.

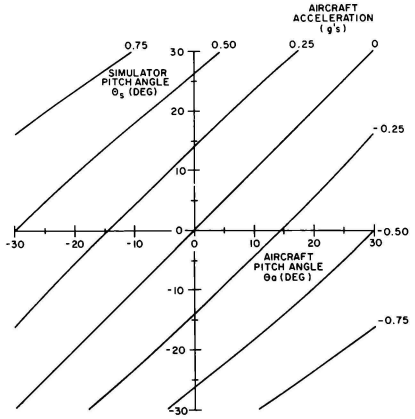


Figure 4. Simulator Tilt Angle, Gravity Vector Alignment Method

False Cues

The gravity vector alignment method assures the same direction of the net force acting through the pilot's body for both simulator and aircraft regardless of the magnitude of the force (or, equivalently, number of g's). But as shown in Figure 2, the number of g's strongly influence the perceived pitch angle which must be matched in the simulator[6], not the line of action of the force. Thus, the gravity vector alignment method does indeed give rise to false cues. Quantitatively, this is described below.

For an aircraft with constant longitudinal acceleration and pitch, Eqs. (6) and (7) reduce to

$$f_x = -\sin(\theta_a + \alpha) - n \cos \alpha \qquad (14)$$

$$f_y = 0 \qquad (15)$$

The perceived pitch is obtained from the zero acceleration condition for which

$$f_x = -\sin(\theta_{a_p} + \alpha) \qquad (16)$$

Equating Eqs. (14) and (16), there results

$$\theta_{a_p} = \sin^{-1}\left[\sin(\theta_a + \alpha) + n \cos \alpha\right] - \alpha \qquad (17)$$

This equation is plotted in Figure 5 for $\alpha = 28.7^\circ$ where it can be seen that the effect of aircraft acceleration on perceived pitch can be significant. For example, accelerating at ½ g at a steady climb angle of 10° results in a perceived pitch of 29°.

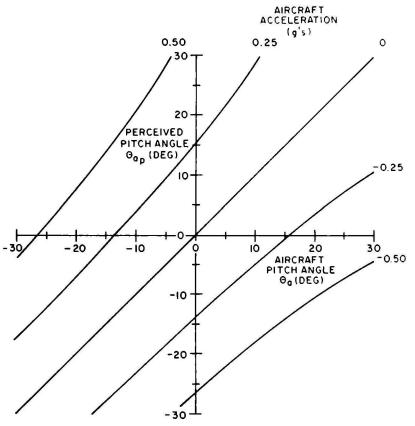


Figure 5. Effect of Aircraft Acceleration on Perceived Pitch Angle

For the simulator tilted through the angle Θ_s by the gravity vector alignment method, the perceived pitch, Θ_{s_p}, is equal to Θ_s since the simulator is in a zero acceleration state for sustained cueing. Therefore, from Eq. (13),

$$\Theta_{s_p} = \Theta_s = \tan^{-1}\left[\frac{n + \sin\Theta_a}{\cos\Theta_a}\right] \qquad (18)$$

From Eqs. (17) and (18), we can compare the pitch perceived in the simulator, Θ_{s_p}, from the gravity vector alignment method with the pitch perceived in the aircraft, Θ_{a_p}.

The following table lists some of these values.

Aircraft Flight Condition		Perceived Pitch in Aircraft	Perceived Pitch in Simulator
n	Θ_a (deg)	Θ_{a_p} (deg)	Θ_{s_p} (deg)
0.25	0	16	14
0.50	-10	21	18
0.50	0	38	27
0.75	-10	49	30

As can be seen, there are some conditions for which the gravity vector alignment method results in perceived pitch comparable to that in the aircraft. However, there are many other conditions for which false cues ($\Theta_{s_p} \neq \Theta_{a_p}$) are significant. The psychophysical significance of the magnitude of these false cues has to be determined separately and will depend on such factors as student tasks being performed, visual cueing, g-seat cueing, and other information (pertinent and extraneous) received by the student.

Modified Gravity Vector Alignment Method

In reference 6, it is shown how to use Eqs. (6) and (7) to simulate sustained motion without introducing false motion cues. In addition, the limits on the aircraft maneuvers and motion platform excursions are evaluated for real and false cue regimes[6]. The basis for quantitatively defining real cues is that the value of the otolith components (f_x, f_y) in the aircraft (Eqs. (6) and (7)) must be the same as that in the simulator[6]. This assures that the perceived sustained motion in the simulator is the same as that in the aircraft.

The method developed in reference 6 should be used for all sustained motion simulation. In that reference, several different degree-of-freedom motion platforms are evaluated. For a platform tilted in pitch, the simulator tilt angle required to provide real cues for simulating an aircraft undergoing climb manuevers is given by[6]

$$\Theta_s = \sin^{-1}\left[\sin(\Theta_a + \alpha) + n\cos\alpha\right] - \alpha \qquad (19)$$

(Note that this equation is the same as Eq. (17) since $\Theta_{s_p} = \Theta_s$ and it is required to have $\Theta_{s_p} = \Theta_{a_p}$.)

This equation represents a modified gravity vector alignment method as contrasted to Eq. (13) used in the conventional approach.

The general tilt orientation in both roll and pitch corresponding to real cues for simulating combined aircraft tilt and acceleration is presented in reference 6. It is this new method which should be used in lieu of or as a modified form of the gravity vector alignment method.

Conclusions

The gravity vector alignment method provides a simple, direct approach to simulating sustained motion cues. However, as demonstrated in this paper, this approach can still lead to false cues. Based on a new theory of tilt perception, the gravity vector alignment method should be replaced by the method of reference 6.

In this new approach[6], the real cue regime is increased considerably; constrained only by the physical excursion limits and by the 1-g limit of a tilted platform simulating sustained motion.

References

1. Parrish, R. V.: "Platform Motion for Fighter Simulations - Let's Be Realistic." AIAA Flight Simulation Technologies Conference, September 18-20, 1978.

2. Schmidt, S. F, and Conrad, B.: "Motion Drive Signals for Piloted Flight Simulators." NASA CR-1601-1970.

3. Conrad, B., and Schmidt, S.F.: "Washout Circuit Design for Multi-Degrees-of-Freedom Moving Base Simulators." AIAA Visual and Motion Simulation Conference, September 10-12, 1973, AIAA Paper No. 73-929.

4. Bitner, M.F : "Investigation of Motion Base Drive Techniques." Naval Air Development Center, Report No. NADC-77306-20, March 1978. AD A053830.

5. Jaslow, H.: "Human Tilt Perception in a Dynamic Environement." Aviation, Space and Environmental Medicine 50(6): 594-598, 1979.

6. Jaslow, H.: "A New Engineering Approach to Motion Cueing Technology." AIAA 18th Aerospace Sciences Meeting, January 14-16, 1980, AIAA Paper No. 80-0047. Journal of Aircraft, 18(3): 220-224, 1981.

7. Jaslow, H.: "Equiperception Maneuvers." Aviation, Space and Environmental Medicine (51(9): 867-871, 1980.

8. Schöne, H.: "On the Role of Gravity in Human Spatial Orientation." Aerospace Medicine 35: 764-772, 1964.

IMPROVED G-CUEING SYSTEM

E.B. Bose
W.P. Leavy
S. Ramachandran
Goodyear Aerospace Corporation
Akron, Ohio 44315

Abstract

A G-Seat employing both hydraulic and pneumatic systems has been designed. The advantages of this design are much reduced response time, availability of both onset and sustained cues, good high-frequency response and sufficient excursions to provide vestibular cues in all degrees of freedom except pure lateral translation. The design also features a microprocessor resulting in: a self-contained G-Cueing System that can be easily fitted into any simulator with minimal changes to the host computational hardware or software; significantly reduced host computer requirements; and a sampling rate of up to 240 Hz without increasing the computational burden on the host simulator. A helmet loader was also designed to supplement the motion cues. The paper presents G-Cueing system design details and pilot evaluation.

Introduction

In recent years, the G-Seat, in conjunction with other G-Cueing devices, has become an accepted means of providing motion cues for combat trainers. While the G-Cueing system cannot replicate the real world accelerations and motions because of the limited physical displacements, sufficient motion cues can be generated to enhance pilot training in the simulators.

Goodyear Aerospace Corporation (GAC) is a pioneer in the development of G-Cueing systems and has been actively pursuing development of motion cueing devices since 1962. The first G-Seat to be built in the USA was the DYNASEAT* (Patent No. 3,270,440) built by Goodyear. More recently, GAC designed a G-Cueing system for the USAF and Royal Saudi Air Force F-15 Flight Simulators.

This paper presents details of the F-15 G-Cueing system design, cueing philosophy, system performance and pilot evaluation. The paper also discusses Goodyear's experience with pilot helmet loaders.

G-Cueing System

The G-Cueing system provides motion cues through a G-Seat, a G-Suit, an active lap belt, and a vibration/buffet system. The conventional G-Seat uses pneumatic cushions in the seat pan, backrest, and thigh panels to provide acceleration sensations to the pilot. Such an all-pneumatic G-seat suffers from several shortcomings such as large response time of up to 350 ms[1] leading to poor correlation with the visual system and an unacceptable system lag error, lack of onset cues, poor high-frequency response, and insufficient vestibular stimulus that could induce sickness in prolonged air combat maneuver training in the presence of strong visual cues.[2] The Goodyear G-seat employs a hydraulically driven seat pan, backrest, and lap belt to overcome the above shortcomings while retaining the sustained acceleration cue capability of the air cushion.

The G-Cueing system consists of the following major components: G-Seat assembly, servo controller rack, fluid power distribution rack, electronic control rack and the fluid power system. A functional block diagram for the G-Cueing system is shown in Figure 1.

G-Seat Assembly

The G-seat assembly (Figure 2) is the most important element of the G-Cueing system. The Goodyear system uses a combination of hydraulic and pneumatic systems to provide motion cues representative of the sensations a pilot feels in an actual aircraft ejection seat while under acceleration loads. The attitude of the seat back and seat pan planes are manipulated by controlling the stroke position of linear hydraulic actuators, and the contacting surface on the trainee pilot's body is controlled by varying the pressure within segmented air cushions built into both the seat pan and back rest.

Some of the components of the G-seat assembly are directly involved with the generation of the cues, while others are ancilliary equipment. These can be classified into categories of active elements, passive elements, and functional elements.

Active elements are components of the G-seat that have been specially designed or adapted to the G-cueing system to produce certain effects or to create conditions peculiar to the simulated environment. Passive elements of the G-cueing system are components that contribute to the realism of the cockpit environment by their presence alone.

Functional elements are components that perform specific functions in the real aircraft cockpit and are incorporated in the simulator to create a more realistic environment. For example, the seat height adjustment which exists in the real aircraft and is duplicated in the simulator, is a functional element. All functional elements must perform identical functions in both the real aircraft and the simulator.

Some elements of the G-cueing system can have a dual role in the G-cueing system. For example, the lap belt is a passive element when it is used

*Registered Trademark of Goodyear Aerospace Corporation

to restrain the pilot in his seat. When the belt is mechanically activated to tense or scrub on the pilot's anatomy, creating a cue, it assumes an active role. Other active elements of the G-cueing system will be discussed later in this paper.

The typical arrangement of the G-seat cueing elements is shown in Figure 3 and consists of a modified aircraft ejection seat and its accessories, an active seat motion assembly, active back rest assembly, active lap belt assembly, passive survival kit belt, functional shoulder harness, and a functional seat-height positioning mechanism.

The active elements of the G-seat assembly are electronically controlled by closed loop servos that are powered by the hydraulic and pneumatic fluid power systems. The primary cueing elements of the G-seat (seat pan, back rest, and lap belt) are controlled linear hydraulic actuators while the secondary cueing elements (seat pan cushion and back rest cushion) are segmented inflatable cells.

The seat pan surface, underneath the segmented inflatable cushion, is driven in three degrees of freedom (heave, roll, and pitch). The motion is provided by three hydraulic linear actuators mounted at right angles to the normal plane of the seat pan with excursions that provide $\pm$ 1.0 inch of heave travel. Roll and pitch angular excursion is maximized within the constraints imposed by the modified aircraft seat.

The hydraulic actuators provide long trouble-free service by use of self-adjusting rod seals and have pistons designed for minimum breakaway friction. The actuator rod ends fit into sliding swivel pads underneath the seat pan surface that accommodate the attachment point shift when the pan has an angular excursion. A centrally located guide post prevents uncontrolled motion of the seat pans in the longitudinal, lateral, or yaw directions while permitting unrestricted motion in the heave, roll, and pitch directions.

Position feedback for each hydraulic actuator is accomplished by using a linear-variable-displacement transducer (LVDT) which has extremely high reliability, long life, infinite resolution, and have no sliding contacts to produce noise, backlash, hysteresis, or wear.

The seat pan motion assembly is designed to be easily removable as shown in Figure 4A. Hydraulic connections use self-locking quick disconnects and the seat pan motion assembly is capable of being repaired on a workbench. No critical mechanical alignment is required.

An inflatable pneumatic cushion constructed of one-inch thick AIRMAT* is mounted on top of the movable seat pan surface. AIRMAT consists of two layers of fabric, impregnated by an elastomer or resin to withstand pressurization, joined by tie threads extending between the upper and lower fabric surfaces. Since the two layers of fabric are restrained by these tie threads, the G-seat cushion contour does not change shape with the level of pressurization, and the tendency for the cushion surface to balloon is controlled by the length of the tie threads. The pneumatic cushion is divided into a four-cell (segment) arrangement and each cell is controlled by air pressure independent of the other three cells. The combination of the pilot weight on the cushion cells and the internal pressure of the cells determines the amount of pilot body area contact and the resulting sensation of firmness. The initial seat pan cell pressure (1-g level) is selected as a function of pilot weight so that all pilots receive similar area contact and firmness sensations.

The inflatable cushion cell provides a heave travel of $\pm$ 0.5 inch. Individual pressure control of the four cells also allows pitch and roll control. Together, the seat pan and inflatable cushion provide total heave travel of $\pm$ 1.50 inches.

Pressure transducers, connected to each of the four inflatable cushion cells monitor the internal cell pressures to provide for individual closed-loop pressure control.

Passive devices in the form of molded pads or blocks, are incorporated into the AIRMAT cushion support plate to provide further pressure stimuli to the buttocks. Additional passive thigh wedges are built into the outboard sides of the seat pan cushion to enhance the body contact sensations to the outer thighs during roll or heave maneuvers. Test results indicate that these devices make the pilot subconsciously aware of lateral cues, even though there is no lateral motion of the seat.

The back rest assembly as shown in Figure 4B is constructed similar to the seat motion assembly. Longitudinal excursions of the back rest linear actuators is $\pm$ 1.0 inch, and the cushion cell excursion is $\pm$ 0.5 inch resulting in a total excursion capability of $\pm$ 1.5 inches. The back rest assembly provides longitudinal yaw, and pitch motion.

Both the back rest and seat pan cushion assemblies are attached to their respective pans by Velcro loop and pile fasteners which are adjustable, reusable, lightweight, vibration resistant, and can be fastened blind with no jamming.

In general, the hydraulic actuators generate onset acceleration cues, and the pneumatic cushions generate sustained acceleration cues. Onset cues are experienced during sudden changes in acceleration (amplitude or direction), and a hydraulic actuator is ideal for providing the required response. The hydraulic actuator's response to a step input command was measured to have a transport delay (to 10% of final value) less than 0.010 sec and a rise time (10-90% of final value) less than 0.015 secs.

In addition to the complex composite cues produced by the aerodynamic equations, vibration and buffet cues can also be introduced through the hydraulic actuators. These vibrations consist of combinations of sine wave signals modulated by pre-programmed envelopes generated by the G-cueing mathematical model. The cueing elements themselves are used to provide the pilot buffet and vibration sensations to frequencies greater than 40 Hz thus eliminating the need for an external seat shaker.

Active Lap Belt Assembly

The active lap belt assembly uses two hydraulic linear actuators, located underneath the seat pan, to position the attachment ends of the lap belt to simulate tightness or looseness of the lap belt.

Differential motion of the lap belt attachments provides additional belt scrubbing cue sensations during lateral and yaw maneuver simulations. The addition of the active lap belt to the G-cueing system allows greater versatility to the G-cueing math model programs so that the lap belt cues can faithfully reproduce those sensations experienced in the aircraft, thus increasing training effectiveness. The procedure for using the lap belt is the same as in the aircraft. The pilot enters the seat, buckles the lap belt, and adjusts the belt straps to the desired tension. During simulated flight, as in actual flight, changes in belt tension may be caused by the pilot's voluntary body movements, but further cue enhancement is provided by additional motion of the belt attachment ends as determined by the G-seat math model programs.

G-Suit

The G-suit, in the real aircraft, is part of the pilot's personal equipment and is used to prevent the rush of blood from the brain that could cause blackouts or tunnel vision during high acceleration conditions. The G-suit as incorporated in the G-cueing system, is used as an active element to provide body pressure discomfort cues that the pilot normally associates with high positive acceleration conditions.

The G-suit servo controller is similar to the G-seat cushion pneumatic controller. Early pilot evaluations indicated that the G-suit was slow to deflate when it was controlled directly by the vertical g-level of the simulated aircraft. This problem was overcome by adding a vacuum system and momentarily forcing the controlling valve to be fully open for deflation based on the sign of the simulated aircraft pitch acceleration.

G-Cueing Processor

A unique feature of the design is the application of microprocessors as part of the G-cueing system to perform all the necessary computations and controls (Fig. 5). The Controller Microprocessor Unit (MPU) is a commercial, fixed instruction, 16 Bit microprocessor designed around a Texas Instrument TM990. This microprocessor serves as an I/O and interrupt controller and provides executive control functions for the entire g cueing system.

The second microprocessor, designated the Arithmetic MPU, is a GAC-developed bipolar, 16 Bit, micro-slice, microprogrammable device that provides the extremely high computation speeds required for the real-time computation application of arithmetic processing. The two microprocessors intercommunicate via shared memory designated as Dual Function Memory. A Direct Memory Access (DMA) card controls the I/O interface between the g cueing system and the simulator main computer. Digital-to-analog (DAC) and analog-to-digital (ADC) conversion cards are used to translate analog and discrete outputs and inputs from and to the g cueing servo circuitry. This circuitry consists of the hydraulic and pneumatic servo amplifier cards, startup/shutdown card, system status card, and a vibration/self test card.

During G-cueing math model development, a program generation memory card was used to store the g cueing model until the pilot evaluation phase was completed, and is shown by the dotted line area in Figure 5. The final g cueing math model is contained in PROMS that are installed in the Arithmetic MPU card which therefore replace the program generation memory card.

Additional PROMS are installed on the Controller MPU card which contain maintenance programs which are written by GAC for the g cueing system. The PROMS contain maintenance and test (M&T) routines. During morning readiness, the M&T routines are run automatically one after another and gives a go/no-go result. During M&T, each routine is executed individually upon command to isolate failures down to the card level.

G-Cueing Methodology

Two very opposing G-cueing philosophies exist in the simulation industry; one is that a positive aircraft g-loading (pull out) should be simulated by deflating the pneumatic seat cells and the other by inflating them. Both approaches were evaluated at GAC and it was generally agreed by most pilot evaluators that it was more effective if the seat got harder (inflated) as the aircraft undergoes positive g's. However, this produced an upward eye-point shift relative to the cockpit displays which is contrary to what takes place in the aircraft. A positive g-loading in the aircraft will produce bending and compression of the spinal column. Hence, the approach adopted was to inflate the seat to provide firmness of the seat in positive g's while simultaneously lowering the seat pan through hydraulic actuators to provide the downward eye-point shift as in the real world. Figure 6 illustrates the approach for generating vertical cues. A similar approach was used for cues in other degrees of freedom also.

The G-cueing algorithm has the following built-in features:

Flexible Software - Since the G-cueing system is a very subjective device, it is essential that the mathematical model can be easily adjusted. Any user who has had experience with motion cueing knows of the importance of the pilot evaluation and tuning process. To accommodate this, coefficients are built into the mathematical model to allow for changing the effects of any stimuli on any seat element independently and/or collectively about any axis. This has proved to be very useful during pilot evaluations and subjective testing.

Acceleration Curve Shaping - In order to facilitate an onset acceleration cue, provisions have been made in the software for shaping the stimuli. However, due to the fast response time of the hydraulic actuators and sensitivity of the seat pan, pilot evaluation revealed that the seat was responsive enough to effectively stimulate even the small changes of control positions and throttles without magnification or shaping.

System Performance

Application of microprocessors as part of the G-cueing system to perform all necessary computations offers the following advantages: a self-contained G-seat that can be easily fitted into any simulator with minimal changes to the host computational hardware and software; development independ-

ent of the simulator offers cost and scheduling advantages; significantly reduced host computer requirements; and a sampling rate of up to 240 Hz without increasing the computational burden on the host simulator. The last capability is quite significant to reduce total system response lag.

A popular method of measuring overall simulator system response lag error (ΔT_{LAG}) is to apply a step input at the pilot stick and measure the time taken for the system to reach 10% of the commanded response and subtract the time taken in the actual aircraft to reach the 10% point. The 10% point represents the time taken to provide onset cues to the pilot. For maximum simulation fidelity and training effectiveness it is desirable to keep this lag error as low as possible.

Figure 7 shows that in addition to the G-seat itself, additional delays are introduced by: 1) the Analog to digital converter (ADC) since it will not be possible to synchronize the ADC with an input that could be randomly applied and, 2) the simulator computer frame time to perform flight calculations. It can be seen that the system lag could be reduced by either reducing the computer frame time (ΔT_{CPU}) or by making the G-seat more responsive (reduce $\Delta T_{G\text{-}seat}$). The first alternative places a great burden on the computation system. Use of hydraulics considerably decreases the system lag since it is faster than pneumatics.

The quick response of the hydraulic servos resulted in roughness of motion cues because of stepping caused by computer iteration sample hold. The solution was to run the microprocessor at 60 Hz and include a 60 Hz notch filter and a 30 Hz low pass filter. However, all these (including the microprocessor) introduce additional system lags. Tests were conducted to determine the effect of the filters and sampling rate on the seat performance and a summary is given in Table 1. The evaluations were done both subjectively and by visually analyzing strip chart recordings. However, since the recorder had a bandwidth of about 100 Hz the sampling effect was not visible above 100 Hz. It may be noted from the Table that by eliminating the filters and increasing the sampling rate to 240 Hz a smooth response could be had with a lag of only 14.2 msecs. The typical lag error for a pneumatic-only G-seat varies from 28 to 350 msecs depending upon the design. [1,5]

Helmet Loader

Goodyear has also investigated the feasibility of providing additional cues through the pilot's helmet. The helmet loader uses dc torque motors, strain gauge beams, and small cables (Figure 8) to exert a force on the head and the shoulders. A military-style flight helmet is implemented so that the force-producing cables are essentially attached to both sides of the helmet through linkages on shoulder straps. Several concepts were tried, but the swivel beam arrangement with the use of suction cups for mounting the device to the helmet and other improvements resulted in less transmission of noise along the force cables, and better alignment of the resultant forces. The small hose shown in Figure 8 was used in experiments to evaluate the need for vacuum assist on the suction cups, and was proven to be unnecessary with the helmet GAC was using.

The force exerted on the head and shoulders is proportional to the aircraft Z axis acceleration. For example, positive g accelerations (pull outs) will produce a "sinking" feeling by weighting down the pilot's head while simultaneously lifting (loosening) the shoulder harness. The electro-mechanical servo loop is a force servo so that the pilot can move his head fore/aft or sideways, with the cable being maintained at a constant tension while being wound or unwound on the drive motor spool.

Pilot Evaluation

The G-cueing system described in this paper, less the helmet loader, is contracted for installation in six F-15 aircraft simulators being manufactured for the Royal Saudi Air Force and the United States Air Force (USAF). The first F-15 trainer with a G-cueing system will undergo final acceptance testing in August 1981.

All preliminary studies and evaluations to date were performed on a GAC laboratory prototype system using the F-15 aircraft aerodynamic math model as a driver. The G-cueing system has been evaluated by USAF and Navy pilots and many non-military personnel. Most of the pilots had experience on a six-degree-of-freedom motion platform simulator, although not all pilots had flown the F-15 aircraft. Their comments and critiques aided in the refinement of the G-cueing math model. All pilots agreed that their evaluations were highly subjective due to a lack of pre-established guidelines and criteria.

Most of the evaluators were impressed with the capability of the G-cueing system in giving them a good jarring. They were complimentary of the excursion range and response capabilities of both the seat pan and back rest. Many of the pilots expressed the opinion that they wanted to get "a good jolt" out of the G-cueing system during aircraft buffeting. Opinions were also expressed that the cues were equal to or more than they had received from a 6 DOF motion system, and more than an all-pneumatic type of G-cueing system.

It was observed that the acceleration evaluations fell into several ranges of g levels. Less experienced pilots tended to limit their maneuvers to the range of 1 to 4 g's. The more experienced pilots, and the F-15 pilots in particular, evaluated the performance of the G-cueing system in maneuvers of 4 to 7 g's. Each group wanted the cues to be exaggerated at their particular range. The more experienced pilots felt that the cues in the lower g ranges should be deemphasized. Several pilots that flew the same mission expressed divergent views on specific cues. Some pilots felt that the roll maneuver of the seat pan should be reduced, and others indicated that an upper body sway should be added. After evaluating and critiqueing the G-cueing system, one pilot admitted that he became more aware of some cues in the real aircraft that he did not notice before. Most recommendations resulted in changes in the G-cueing math model leading to improved performance characteristics.

Before the demonstration rides, some of the

evaluators had the preconceived notion that the G-cueing system could not be used to replace a seat shaker type vibration/buffet system. After the demos, most were convinced that they had been wrong. The unanimous conclusion shared by the evaluators was that the G-cueing system hydraulic actuation system provides excellent vibration, buffeting, and turbulence simulation without the necessity of providing an external seat shaker. This conclusion was also observed by Albery [3,4] in the tests at Human Resources Laboratory (HRL), Wright Patterson Air Force Base.

Another unanimous opinion expressed was that the active lap belt was a very effective cueing device. There was a difference of opinion as to whether it was more effective to loosen, or tighten the belt during the positive g maneuver and these will be investigated further during the in-house Pilot Evaluations and Acceptance Tests. McGuire[1] has reported the same results during HRL tests at Williams AFB.

Another conclusion reached as a result of the many demonstrations was that it was very important that the anti-G suit properly fit the pilot. It was noted that improperly fitted G-suits could prevent effective evaluation of the cueing sensations provided by the G-suit math model. Many of the evaluators commented that a slow initial inflation of the G-suit could be directly attributed to the fit of the G-suit. This conclusion supports the necessity of providing controls at the Instructor Station to allow adjustment of the point of onset of suit inflation and slope of the suit pressure versus g's. By adjustment of these controls, proper compensations can be made to tailor the G-suit performance to the individual. This problem will probably be minimized in an actual trainer situation since each pilot will be wearing his own personal G-suit.

Pilot evaluation included the helmet loader also. The pilots expressed the opinion that the fatigue in the neck and shoulder muscles normally experienced in high g aircraft maneuvering over long periods was evident with the addition of the helmet loader to the g cueing system simulation.

There was a consensus amongst the pilots that the effectiveness or the level of the Z axis acceleration cue was more pronounced through the helmet loader than through the g cueing seat elements. This could be attributed to the fact that the force on the head results in compression of the neck, shoulders, and spine which is not provided by the g cueing seat alone.

Most of the evaluators were surprised to find that they quickly became unaware of the connecting cables and that the cables didn't restrict head motion as they had expected.

Conclusion

A G-cueing system that employs hydraulics for onset cues and pneumatics for sustained cues was designed. This design provides several advantages over an all pneumatic system such as: greatly reduced response time, availability of onset cues, good high frequency response and large excursions to provide vestibular cues. Pilot's subjective evaluation and their enthusiastic response to the G-cueing system tend to confirm this.

The helmet loader could be very effective in supplementing the motion cues if it can win pilot acceptance. However, presently, lack of enough information on the effectiveness of this device and the psychological resistance to any deviations in the trainer cockpit to the real world environment have been the main factors for the lack of acceptance. The Military Research and Development Commands must play an active role in promoting the acceptance of the helmet loader amongst the training commands.

References

1. R.L. Engel, D.C. McGuire: "ASPT G-Seat/G-Suit Optimization," Proceedings of Interservice/Industry Equipment Conference, November, 1980.

2. E.A. Stark: "Motion Perception and Terrain Visual Cues in Air Combat Simulation," Proceedings of AIAA Visual and Motion Simulation Conference, Dayton, Ohio, April 1976.

3. W. Albery: "ALCOG Film and Narrative," AFHRL, Dec. 1977.

4. G. Kron, L. Young and W. Albery: "High G Simulation - The Tactical Aircraft Simulation Problem," Proceedings of the 10th NTEC/Industry Conference, Nov. 1977.

5. B.R. Ashworth, et al: "Objective and Subjective Evaluation of the Effects of a G-seat on Pilot/Simulator Performance During a Tracking Task," Proceedings of the 10th NTEC/Industry Conference, November, 1977.

Table 1. Subjective Evaluation of G-Seat

Sample Freq. Hz.	Notch Filter	Low Pass Filter	Cue Roughness	*Acceptability (Scale 1-10)	G-Seat Lag ($\Delta T_{G\text{-}Seat}$)
60	In	In	Barely Discernible	8	37.1
60	Out	In	Discernible, Somewhat Objectionable	5	33.9
60	Out	Out	Very Objectionable	1	26.7
120	Out	In	Not Discernible	10	25.53
120	Out	Out	Objectionable	4	18.33
180	Out	Out	Objectionable	5	15.56
220	Out	Out	Threshold of Detection	9	14.55
240	Out	Out	Not Discernible	10	14.2

* Acceptability was based on relative sensing of roughness; 10 on the scale was considered equal to operation without sampling.

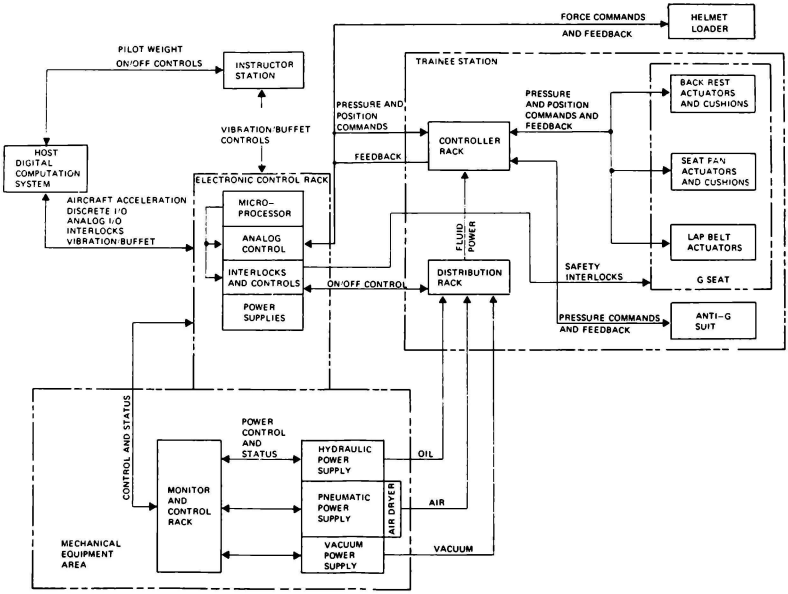


Figure 1. Functional Block Diagram of G Cueing System

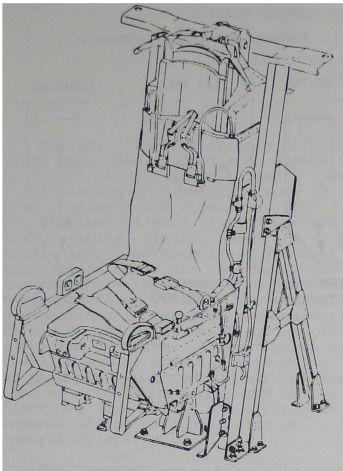


Figure 2. G-Seat Assembly

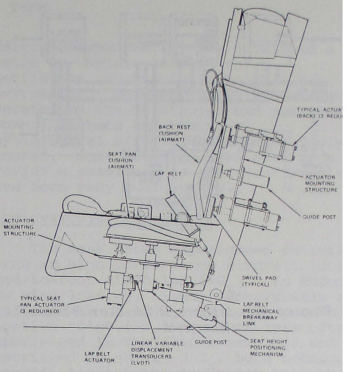


Figure 3. Typical Arrangement of G Seat Cueing Elements

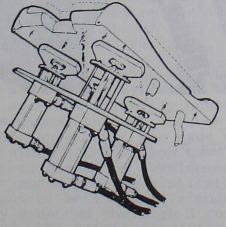


Figure 4-A. Seat Motion Assembly

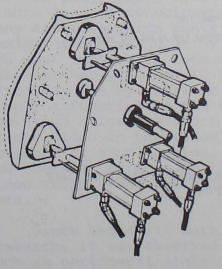


Figure 4-B. Back Motion Assembly

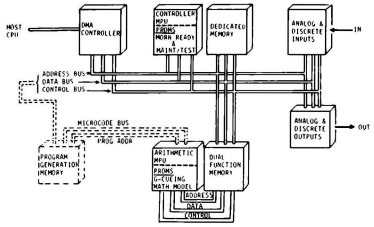


Figure 5. Microcontroller Block Diagram

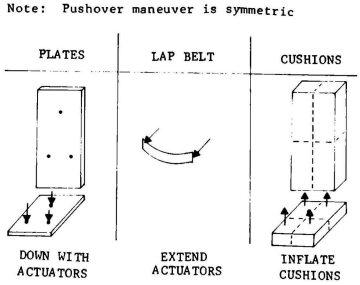


Figure 6. Cueing Philosophy for Pull Out (Positive g's)

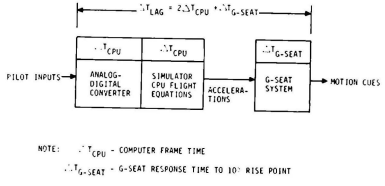


Figure 7. System Worst Case Lag Error

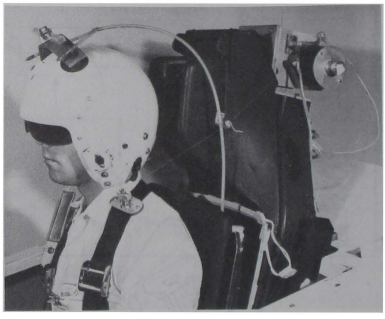


Figure 8. Helmet Loader

PHYSIOLOGICAL EFFECTS OF HIGH-G FLIGHT THEIR IMPACT ON FLIGHT SIMULATOR DESIGN

F. M. Cardullo*
State University of New York
Binghamton, New York

Abstract

The physiological effects of high G flight have been studied and related to the flight simulator environment. Means of stimulating or simulating some of these phenomena are discussed. The physiological effects treated include cardiovascular, vision, respiratory, tactile and proprioceptive. The most important of these are the cardiovascular since they are the genesis of the well known effects on vision. These effects were examined in terms of the geometry of the diminution of visual field and the associated dynamics. A description of several means of creating this effect in the simulator is presented. These include means of actually inducing the visual effects by stimulating the appropriate response of the cardiovascular or respiratory system.

The research upon which this paper is based was conducted by the author together with Professor Laurence R. Young of the Massachusetts Institute of Technology and Mr. Gerald J. Kron of the Link Division, Singer Company. The work was sponsored by the U. S. Air Force Human Resources Laboratory.

Introduction

The same well known reasons that have been responsible for the increased utilization of flight simulators for training pilots, also motivate the heightening interest in increasing the amount of ACM (Air Combat Maneuvering) training to be provided in flight simulators. Therefore, it becomes necessary to provide more of the physical cues from that environment in the sumulator.

Initially, acceleration cues were provided by moving base systems. These, in general, provided primarily onset cues and, in some cases, through the use of gravity align, an attempt was made to provide sustained acceleration cues. In the more recent past, the development of the "G-Seat" and a more sophisticated implementation of anti G-suit simulation have greatly enhanced the high G cuing environment in the flight simulator. In addition, Ashworth (1) has developed a system for helmet loading which contributes substantially to this cue environment. However, these systems have operated on the vestibular and somatic sensory systems and the other physiological effects have been largely ignored. Some attempts have been made to provide a semblance of "blackout" but these have been very rudimentary.

*Assistant Professor, Technical Studies
Member AIAA

Cardiovascular Effects

The most often experienced inertial force in high performance aircraft is in a head-to-foot direction ($+G_z$). In modern low wing loading, highly maneuverable aircraft, levels of 12 G_z are quite possible. Negative "G's" ($-G_z$) are much less often experienced in the flight envelope and are restricted to about -3 G_z, both because of aircraft limitations and pilot tolerances. Longitudinal acceleration $\pm G_x$ are confined to the range of ± 2.5 G_x but are usually in the range of ± 1 G_x unless aircraft carrier operations, of catapult launch and arresting, are considered. In the lateral directions ± 1.5 G_y is generally the maximum capability while the usual environment is on the order of 1/4 G_y.

High levels of $+G_z$ (resulting from aircraft acceleration upward) during aircraft meneuvers have the most profound effect on the cardiovascular system. The circulatory effects of this flight regime have been recognized since the early days of acceleration research because they result in progressive loss of vision and ultimate loss of consciousness. There are many elements which contribute to the dynamics of the vascular system. However, the most important, from the viewpoint of interest here, is due to the hydrostatic effects. The hydrostatic pressure is simply a function of the vertical distance above and below the heart and the weight of the blood in the vessels. In the presence of acceleration, it is primarily the inertial effects on the mass of the blood which cause the effects of interest here.

Since the early days of acceleration research, it was observed that symptoms such as peripheral light loss, to blackout, to loss of consciousness could only be explained by a decrease in blood flow to the head. A simplistic view of the cardiovascular phenomena associated with G_z shows that as G_z increases, the hydrostatic pressure in the legs increaes, the vessels passively dilate, and a major portion of the blood from the upper part

of the body is displaced to the lower vessels. Since the venous return decreases, the cardiac output decreases thereby further causing a decrease in pressure at the aorta and above the heart. A more rigorous explanation involves a discussion of initial inertial effects, hydrostatic effects, orthostatic effects and the reflex activity of the cardiovascular and central nervous system.

Retinal perfusion requires an eye level arterial pressure in excess of approximately 20 mm Hg, independent of reduced venous pressure, to overcome intraocular pressure and keep the arteries and veins from collapsing. If the eye level blood pressure falls below this, diminution of vision and eventual blackout will occur. Loss of vision precedes unconsciousness because cerebral flow is still adequate. Visual failure is a continuum from loss of peripheral vision (grayout) to total loss of vision (blackout).

While $+G_z$ accelerations are the most often encountered and produce the most profound cardiovascular effects, $-G_z$ accelerations produce some significant cardiovascular effects. Negative G_z is an acceleration which produces a foot-to-head inertial force. The effects are the opposite of what occurs in $+G_z$. Blood from the lower part of the body drains towards the head causing the soft tissue of the face and neck to become engorged with blood. The engorgement may cause petechial hemorrhages over the conjunctiva. This phenomenon may or may not cause the so called "redout". This $-G_z$ effect is controversial. The blood engorgement produces a full feeling in the head and pain in the face and eyes. Transverse acceleration produces virtually no cardiovascular effects in the range of G_x experienced by modern tactical jet aircraft.

Visual Effects

The major effect of high G flight is the loss of visual acuity. This phenomenon limits the pilot's ability to take maximum advantage of his aircraft's performance. If the pilot "blacks out" he must break off an engagement or at least reduce the number of G that he is "pulling" until his acuity returns. It should be noted at this point that "blacking out" refers only to loss of vision and not to unconsciousness. Virtually all visual effects arise from cardiovascular effects. In the Z axis the major problem is either not enough blood at the retina ($+G_z$) or too much blood at the retina ($-G_z$).

In order to understand the manner in which vision is affected by the high acceleration environment it is important to appreciate the distribution of blood throughout the retina. After the main artery enters the eyeball at the optic disc, it divides and subdivides into increasingly fine vessels as they approach the periphery of the retina, resulting in a blood pressure reduction as the vessels decrease in size. If the blood pressure is reduced at the source, it follows logically that a reduction in blood supply would occur initially at the periphery and then proceed inward as the pressure drops further. Hence, concurrent with a loss of pressure there is a loss of visual acuity. This phenomenon when precipitated by a reduction in systemic pressure due to $+G_z$ acceleration, is often referred to as "tunnel vision". Figure 1 provides an illustration of the manner in which the visual field collapses as a function of acceleration. This figure presents the visual field for the right eye. Note the nasal degradation initially more pronounced and then the temporal loss. The elliptical area to the right of the fixation point (the center of the diagram) is the optical disc or "blind spot". The pilot observes this phenomenon as initially a gradual loss of peripheral vision (grayout) then ultimately total loss of vision (blackout). In order to provide a relationship between G_z and the acuity of vision at any point in the field, a mathematical model of the physiological phenomena involved could be developed.

Some notes on negative G_z are appropriate at this point. In general, a small percentage of aircraft maneuvering is accomplished in the negative G environment. This is a result of two factors: 1) aircraft are not stressed to accommodate this environment and 2) pilots are uncomfortable in this environment and, therefore, avoid it. The most controversial area of negative G_z is that associated with the "redout" phenomenon. This phenomenon has had a great mystique associated with it.

The mechanism which causes redout is also controversial. Some postulate petechial hemorrhages proliferated by lacrimal fluid; others conjecture that the engorged conjunctiva are pulled over the eyes. The visual effects of transverse acceleration are minimal. The effects that do occur are at high levels of sustained G_x 12 g and these are primarily blurring and excessive lacrimation due to distortion of the eyeball.

Four techniques are suggested to induce the diminution of vision due to the high acceleration environment. The first employ lower body negative pressure (LBNP) in order to stimulate the cardiovascular effects previously mentioned, thereby causing the visual effects resulting from less than adequate retinal perfusion. Secondly, principles of ophthalmodynamometry are employed to reduce the blood flow to the retina. The third approach alters the in cockpit and out-the-window visual scene to simulate the effects of tunnel vision. Finally, a technique employing

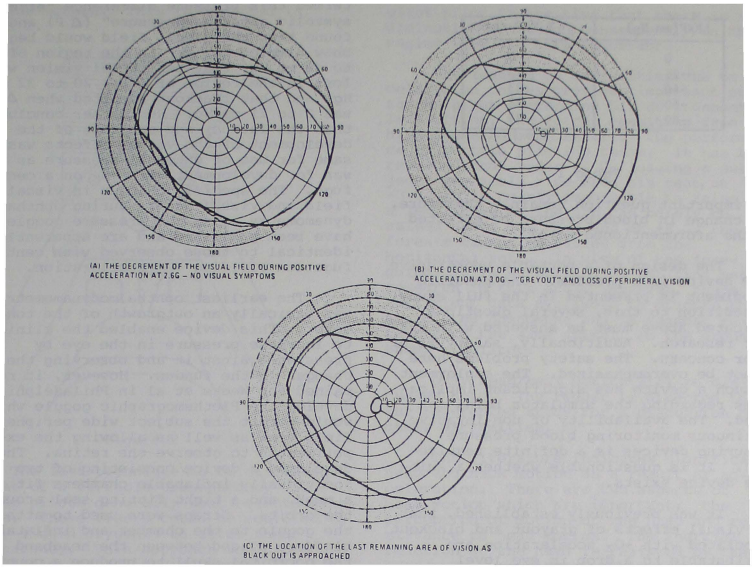


Figure 1

liquid crystal technology is suggested to vary the transparancy of the pilot's face mask.

Experiments involving reduced pressure to various parts of the body have been conducted since about 1841. These first experiments applied reduced pressure to a small anatomical area, to effect localized hyperemia, thereby drawing blood away from diseased organs or areas of the body. Subsequently larger areas were subjected to reduced pressure and it was noted that the resultant pooling in the extremities could induce syncope, which at the time was a satisfactory state for surgery.

Subsequently, lower body negative pressure (LBNP), where the reduced pressure is applied from the iliac crest down, was used to conduct research in cardiovascular deconditioning in bed-ridden patients. Most recently LBNP was used in the Skylab spacecraft to counter the effects of cardiovascular deconditioning of the crew due to long periods of weightlessness. In 1976, J. C. Howard first suggested the use of LBNP in a flight simulator. He theorized that the visual effects of high G flight could be induced in a ground based flight simulator by artificially reducing venous return from the legs by employing LBNP. To this date no simulator related LBNP research has been conducted. Therefore, data collected for other purposes must be used in order to predict the viability of Howard's hypothesis. One of the major shortcomings of the available data is the lack of short time dynamics. It is known that after 10 minutes, there is significant reduction in blood pressure, but the short term effects are not clear. Perhaps larger levels of LBNP would produce more significant results. Other data indicate that -80 mm Hg is sufficient to induce symptoms of impending syncope. It also shows that at that level of LBNP there is a 37% reduction in systolic pressure (Table 1). Based on the fact that a nominal level for central light loss is 5 g then one may assume that -80 mm Hg is approximately the amount of LBNP required for simulation of 5 g maneuver. Since an aircraft is capable of 10 g/sec rate of change of acceleration, then the response of LBNP produced reduction in blood pressure would have to be 180 mm Hg/sec.

LBNP(mm Hg)	% Reduction
0	0
-25	12
-40	23
-60	32
-80	37

Table 1

The important question remains, therefore, can change in blood pressure be effected at the aforementioned rate?

The design of a portable unobtrusive LBNP device is a problem. One possible embodiment is presented in the full study. In addition to this, several questions indicated above must be answered with further research. Additionally, safety is of major concern. The safety problems here cannot be overemphasized. The employment of such a device has significant implications regarding the simulator being man rated. The availability of non-invasive continuous monitoring blood pressure measuring devices is a definite requirement. It is questionable whether a suitable device exists.

It was previously established, that the visual effects of grayout and blackout associated with $+G_Z$ acceleration are attributable to a drop in eye level arterial pressure. When pressure in the retinal arteries drops to below intraocular pressure (P_i) visual impairment results in a matter of a few seconds. It remained for a series of excellent experiments using ophthalmodynomometry, which artificially raised the intraocular pressure to reach or exceed that of the retinal artery, in order to demonstrate that the visual effects under both $+G_Z$ and increased P_i were identical, and that each was attributable to the original.

As will be discussed below, all of the visual effects associated with $+G_Z$ can be reproduced in a one G field using plethysmographic goggles which artificially raise P_i by applying greater than ambient pressure air to the eyeball. Experiments with such goggles and centrifuge accelerations have been able to demonstrate conclusively the anatomical site of the visual phenomenon of grayout and blackout.

E. H. Lambert, in a brief report in 1945, was the first to demonstrate directly that the physiological basis of acceleration blackout could be explained and mimicked by raising intraocular pressure through ophthalmodynomometry. He concluded that the visual effects of acceleration and of externally applied pressure depended only upon the difference between systemic blood pressure at eye level and any externally applied pressure, as through plethsmographic goggles. He termed this pressure difference "effective systolic arterial pressure" (Δ P) and found that the visual field would begin to show dimming for ΔP in the region of 30 to 49 mm Hg, that peripheral vision was lost in the region of Δ P = 20 to 32 mm Hg, and that blackout resulted when Δ P was 0 to 21 mm Hg. He further concluded that the latency and progress of the development of the visual effects was the same for use of external pressure as it was for development with $+G_Z$ on a centrifuge. The specific changes in visual field and visual acuity during ophthalmodynamometry using the pressure goggles have been published and are apparently identical to those observed with centrifugation yielding $+G_Z$ acceleration.

The earliest ophthalmodynamometry was basically an outgrowth of the tonometer. This device enabled the clinician to increase pressure in the eye by pressing against it and observing the changes in the fundus. However, it remained for Weeks et al in Philadelphia to develop the Plethsmographic goggle which would permit the subject wide peripheral field view as well as allowing the experimenter to observe the retina. They developed a device consisting of two individually inflatable chambers fit into a mask, and a tight fitting seal around the orbits. Straps were used to attach the goggle to the chamber and inflatable bladders placed between the headband and the occipital skull to produce a counter-pressure for reduced discomfort and better fit. These goggles could be pressurized to 200 mm Hg. It was found that grayout occurs at approximately 8 to 25 mm Hg above eye level diastolic pressure.

Several modifications to the above device have been attempted but the results still fall short of meeting the requirements for flight simulation. However, the approach does show promise and is inexpensive.

The third technique mentioned is that of altering the pilot's view of the instruments and out-the-window image to simulate the effects of tunnel vision. Several attempts at this have been tried previously but have not met with a great deal of success. As indicated previously, the pilot's vision is reduced during exposure to high $+G_Z$. The geometry of this degradation of vision is fairly complex. Consequently, it seems appropriate to simulate the tunneling via concentric circles. These circles are centered on the line-of-sight as established by an oculometer. This is one aspect that has been previously neglected and may explain the previous lack of success. At this point, two separate algorithms are required, one to determine which instruments are to be dimmed, the other determines the degradation of the out-the-window display. One of the more difficult problems is blending the interior and

exterior scenes. This is depicted in Figure 2.

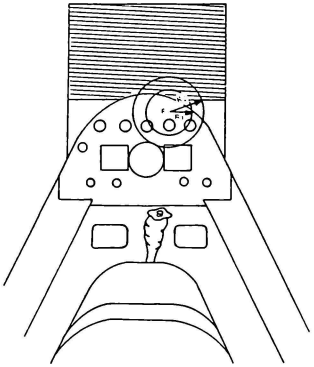


Figure 2

The last technique to be discussed is that of a variable transparancy visor. Fighter pilots generally wear helmets with visors. These visors could be modified to employ liquid crystals to alter their transparancy. This approach would produce very little loss of environmental fidelity. The pilot's instantaneous line of sight is constantly monitored by an oculometer and the tunneling effect is centered about this line of sight. The approach here is to employ liquid crystal techniques to vary the visor transmittance factor. If the liquid crystals are sandwiched between two transparent conductors, the visor, then the optical characteristics of these liquid crystals may be altered by variation of the applied voltage. This technique shows a great deal of promise, primarily because it does not impose any restrictive paraphernalia on the pilot and solves many of the interface problems involved in the third approach. Some technological breakthroughs are required, however. One is constructing a "sandwich" of liquid crystals in a curved rather than flat plate visor. The other is the ability to matrix address the various liquid crystal elements.

Musculoskeletal Effects

The musculoskeletal effects are divided into three categories. They are; the reaction of the extremities, the reaction of the head and neck and the helmet relative to head motion. These effects are very important in the hierarchy of stimuli as reported by subjects exposed to the high G environment. Together with the tactile stimuli, they rank next in importance to the visual effects. However, these stimuli provide cues to the pilot prior to the time that there is any diminution of vision, consequently in that regime they are more important.

The loading of extremities due to acceleration forces provides important cues to the pilot concerning his environment. In addition, the forces resulting from the high G environment restrict the performance of certain piloting tasks. It has been reported that under +2 G_z loading a subject can barely rise from his seat, at +3 G_z it is nearly impossible to raise a leg, at +6 G_z the arm can barely be raised above the head and at +8 G_z the forearm cannot be raised from the horizontal rest. In view of the foregoing, it is surprising to not find more research in this area. However, there are sufficient data available to discuss the associated phenomena and to suggest means to create the same stimuli in the ground based simulator.

Only head/neck motion will be discussed here. The reader is directed to the referenced study for a discussion of other extremities.

The literature indicates significant head and neck motion as a result of acceleration. There are two aspects of head motion which are important; the perception of motion of the head and resistance to head motion. The inertial reaction of the head is resisted by the head/neck musculature. These muscles not only resist motion caused by external forces, but they also possess the sensors which detect external displacement forces. These sensors are of the same type as indicated above in the extremities discussion. The head/neck system has been modeled by several authors, including Don Gum, as an inverted pendulum. The output of Gum's model is the muscle spindle afferent firing rate in response to a specific force input. This model could be used as a tool in the design of a head/neck stimulation system. In addition to the muscular sensors, head motion is sensed through the vestibular apparatus. Data for head/neck motion response to G_z are available. These data indicate that head pitch is most significant motion in terms of excursion. The data show a considerable variability among the subjects. There appears to be a trend for head pitch to overshoot the 1 g position upon return. Head movement seems to be independent of helmet weight. The data show that there is no significant rolling or yawing of the head due to G_z but an average 2° pitch down of the head at 6 G_z. This motion is shown to be linear from 1.0 to 6.0 G_z. It was found that the eye point was depressed 50 mm under 6 G_z; 1/6 of this was due to head pitch, the remainder neck/torso compression. This motion was also found to be linear from 1.0 to 6.0 G_z.

One means of inducing these effects in a simulator is through the helmet loader reduced to practice by Billy Ashworth at NASA Langley. The system employs two small cables attached to the helmet and through pulleys on the shoulder harness to a torque motor and employing a force transducer in the control loop. The force feedback is utilized in order to follow the pilot's movements while maintaining the appropriate helmet forces. Figure 3 shows the installation of the NASA helmet loader in the DMS at Langley. The two small pulleys attached to the pilot's shoulder harness provide a loosening of the straps as force is exerted downward on the helmet for positive G. Unrestricted head movement is permitted by the amount of cable wound on the reel of the torque motor.

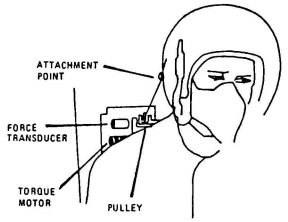


Figure 3

This device is scaled to produce 2/3 of the inflight helmet loads up to a maximum of 6g where the total force exerted is 9 pounds. The developers state that their studies on the DMS have shown that the "helmet loader has a measureable effect on the pilot/simulator system". They further state that "subjective data indicates that the cue provided through the use of the helmet loader is realistic, and there is no noticeable time delay in the presentation of the cue. However, the pilots had mixed opinions about the effect of the helmet on their performance." The device is designed with pilot safety as a consideration and thereby uses breakaway snaps on the helmet, current and voltage limiting and small torque motors to ensure that the pilot does not experience excess forces. One disadvantage to this system is that it does not lend itself to inducing head pitch down, resulting from $+G_Z$. However, with the addition of one torque motor and a helmet attach point at the base of the helmet, pitch up could be achieved as would be expected in $+G_X$ maneuvers. Although the data does not necessarily support this modality, it appears that there should be head motion in this degree of freedom.

Tactile Stimulation

Tactile sensations, i.e. the sense of touch and pressure, have become recognized as important cues to a pilot in discriminating his acceleratory environment. Tactile stimuli are presented to the pilot of an aircraft maneuvering under high G by several means. He senses force changes in the buttocks, lap belt and shoulder harness regions as a result of changes in G_Z. He senses movement of the oxygen system face mask and his back relative to the seat. The effects can be broken down into two categories, those acting normal to the skin and those acting tangential to the skin. The latter can be thought of as "scrubbing" of the skin against some object, e.g. a face mask, lap belt or seat back.

In the past, the main impetus of motion simulation has been concerned with stimulating the vestibular system. It has only been within the last ten years that any significance has been given to the importance of stimulating the haptic system. This recognition led to the development of G-seat systems and improvement of anti-G-suit system simulation. The question becomes; is this enough? There is a paucity of test data regarding the response of the various tactile sensors to the accelerated flight regime. One conclusion that is intuitive is that the intensity of the stimulation of these sensors increases with increased acceleration levels. However, this does not provide a functional relationship with which the hardware used to provide these cues could be driven.

In terms of theoretical information, some work has been done to model the pressure sensors. However, very little has been done in the area of skin tension or scrubbing sensations. Therefore, the answer to the question posed above is not available in a concrete sense. Consequently, the addition of other devices used to stimulate these receptor systems must be justified on the same basis that much simulation hardware has been justified previously; i.e. if the pilot uses the information provided by these sensors, then the stimuli should be present in the simulator. Unfortunately, this then becomes a very subjective basis for simulator requirements.

A more thorough description of the haptic system may be found in the referenced study and its bibliography. Several approaches are readily available for stimulating the tactile receptors. One is by adjusting shoulder harness tension and another is to impose a downward force on the oxygen mask. This latter stimulus is one of the most frequently pilot reported stimuli.

Respiratory Effects

As was stated previously, the dominant physiological effects of high G flight are cardiovascular in origin. It has been found that even moderate levels of prolonged headward acceleration can cause noticeable symptoms because of reduced oxygen transport. The effects on respiration are much less dramatic than the cardiovascular ones for G_z. However, this situation is reversed for G_x. In forward acceleration difficulties in breathing are noticed long before any cardiovascular problems become evident. The inertial load on the chest causes an increase in the work required to expand the chest, thereby leading to shallow breathing. This combines with a significant increase in the effective dead space of breathing and a major mismatch between perfusion and ventilation in the lung region, to make gas exchange inefficient and to cause hypoxia even with inspired O_2. With the advent of tilt back seats to lessen the cardiovascular effects of G_z, the component of acceleration in the X-physiological axis increases and, therefore becomes more important.

While the respiratory effects due to G_z are minimal as compared with the cardiovascular effects, it is worthwhile to mention them. Inspiration becomes difficult and there is a reported tendency for breath to be held. Also, O_2 saturation drops. For G_x the respiratory effects are of major concern. Chest tightness and pain along with difficulty of breathing are reported at about $+5G_x$ and increase with further acceleration until at about $+12G_x$ breathing difficulty and chest pain are severe.

One approach for providing these cues could be implemented via the "subatmospheric face mask". This implementation would simply reduce the pressure of air in the face mask causing the pilot to breathe more rapidly in order to satisfy his oxygen deficit. Negative pressure breathing produces similar effects as does G_x. Therefore, it forms the physiological basis for the subatmospheric face mask. The subatmospheric face mask would provide breathing air to the pilot at pressures below atmospheric. This can be accomplished by employing a system as shown in Figure 4. For accelerations less than or equal to 1.0g, the mask is supplied air at 0.05 psi and for accelerations in excess of 1.0g, the pressure is reduced .096 psi/g. This will cause the breathing rate to increase probably producing some chest tightness and pain. The safety of the system can be assured by the respiration rate being fed simultaneously into the control electronics whereupon the solenoid is shifted to the pressure breathing side if the respiration rate falls below some threshold such as 2 breaths/minute. In addition to the face mask can be easily removed by the pilot if he feels distressed.

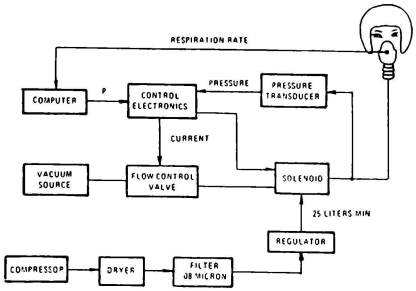


Figure 4

CONCLUSION

Several of the physiological effects of high G flight have been briefly described and means of either simulating or stimulating these effects have been discussed.

It was stated that the most significant effects are those on vision and that they have their origin in the cardiovascular response to high G stress. Two means of stimulating and two means of simulating these vision effects have been presented. In addition, musculoskeletal, tactile and respiratory system stimuli were described and at least one means for producing each of these effects was presented.

Some of the techniques introduced require more research to demonstrate their feasibility, some are well within the state of the art and need only be designed while others, such as Ashworth's helmet loader, can be implemented directly. It is the hope that this paper will stimulate continued research into this area of flight simulation.

REFERENCES

1. Ashworth, B. R., McKissick, B. T., "The Effect of Helmet Loader G-Cuing on Pilot's Simulator Performance". AIAA Paper 78-1573, (1978).

2. Kron, G. J., Cardullo, F. M., Young, L. R., "Study and Design of High G Augmentation Devices for Flight Simulators" Report F33615-77-C-0055, Jan. 1980. Prepared for Human Resources Laboratory Wright-Patterson Air Force Base, Ohio.

3. Gum, D. R., "Modeling of the Human Force and Motion Sensing Mechanisms". AFHRL-TR-72-54, (1973).